Oracle® Fusion Applications

Enterprise Deployment Guide

11*g* Release 1 (11.1.1.5)

E16684-02

September 2011

ORACLE®

Oracle Fusion Applications Enterprise Deployment Guide, 11*g* Release 1 (11.1.1.5)

E16684-02

Primary Author: Karen Ram

Contributors: Janga Aliminati (architect), Sriramulu Lakkaraju, Sebastien Arsenne, Christelle Balon, Faouzia el Idrissi, Susan Kornberg, Pascal Prevot, Anil Ranka

Contents

Preface xi

Audience xi
Documentation Accessibility xi
Related Documents xi
Conventions xii

What's New in This Guide for Release 11.1.1.5 (September) xiii

Documentation Changes for Release 11.1.1.5 (September) xiii

1 Enterprise Deployment Overview

1.1 What is an Enterprise Deployment? 1-1
1.2 About Oracle Fusion Applications 1-2
1.3 Benefits of Oracle Recommendations 1-3
1.3.1 Built-in Security 1-3
1.3.2 High Availability 1-3
1.4 Terminology 1-4
1.5 Reference Enterprise Deployment Topology 1-7
1.5.1 Overall Reference Enterprise Deployment Topology 1-7
1.5.2 Oracle Web Tier 1-9
1.5.3 Application Tier 1-10
1.5.4 Data Tier 1-11
1.6 Hardware Requirements 1-11
1.7 Installation Prerequisite 1-11
1.8 Implementing the Topology 1-12

2 Network Configuration

2.1 External Virtual Server Name 2-1
2.2 Internal Virtual IP 2-2
2.3 Load Balancer Configuration 2-2
2.4 Reference Enterprise Deployment Directory Structure 2-2
2.4.1 Directory Structure 2-3
2.4.2 Binary Directory Structure 2-4
2.4.3 Domain Configuration Directory Structure 2-5
2.5 Shared Storage 2-5
2.5.1 Shared Storage for Oracle Business Intelligence 2-5

2.6 IPs and Virtual IPs 2-6
2.7 Firewalls and Ports 2-8
2.8 Clock Synchronization 2-11

3 Setting Up the Database Tier

3.1 Understanding the Database in the Enterprise Deployment Topology 3-1
3.2 Setting Up the Database 3-2
3.2.1 Database Host Requirements 3-2
3.2.2 Supported Database Versions 3-2
3.2.3 Minimum Database Configuration Parameters 3-3
3.3 Creating and Starting the Database Services 3-5
3.3.1 Updating the Kernel Parameters 3-5
3.4 Loading the Oracle Fusion Applications Repository into the Oracle RAC Database 3-6
3.5 Backing Up the Database 3-11

4 Using the Provisioning Process to Install Components for an Enterprise Deployment

4.1 Understanding Provisioning 4-1
4.2 Prerequisites for Using the Provisioning Process 4-1
4.3 Installing Components 4-3
4.3.1 Creating the Installation Environment 4-3
4.3.1.1 Downloading the Provisioning Repository 4-3
4.3.1.2 Installing the Provisioning Framework Bits 4-3
4.3.2 Creating a New Provisioning Plan 4-5
4.3.2.1 Installation Options Screen 4-5
4.3.2.2 Specify Security Updates Screen 4-6
4.3.2.3 Provisioning Configurations Screen 4-6
4.3.2.4 Plan Description Screen 4-7
4.3.2.5 Installation Location Screen 4-7
4.3.2.6 System Port Allocation Screen 4-9
4.3.2.7 Database Configuration Screen 4-10
4.3.2.8 Schema Passwords Screen 4-11
4.3.2.9 ODI Password Configuration Screen 4-12
4.3.2.10 Domain Topology Configuration Screen 4-13
4.3.2.11 Web Tier Configuration Screen 4-14
4.3.2.12 Virtual Hosts Configuration Screen 4-15
4.3.2.13 Load Balancer Configuration Screen 4-16
4.3.2.14 Web Proxy Configuration Screen 4-17
4.3.2.15 Identity Management Configuration Screen 4-18
4.3.2.16 Access and Policy Management Configuration Screen 4-21
4.3.2.17 IDM Configuration Screen 4-23
4.3.2.18 Summary Screen 4-23
4.3.3 Running the Provisioning Commands to Install Components 4-24
4.4 Configuring Components 4-25
4.5 Performing Post-Provisioning Validation 4-28

5 Scaling Out Oracle HTTP Server

5.1 Performing the Scaleout 5-1
5.2 Installing WebGate Patches 5-24
5.3 Wiring Oracle HTTP Server with Load Balancer 5-24
5.4 Validating Oracle HTTP Server on WEBHOST2 5-25

6 Configuring Node Manager

6.1 Configuring Node Manager for CRMHOST2 6-1
6.2 Creating the Identity Keystore on CRMHOST2 6-2

7 Scaling Out the Oracle Fusion Customer Relationship Management Domain

7.1 Overview of the Oracle Fusion Customer Relationship Management Domain 7-1
7.2 Prerequisites for Scaling Out the Oracle Fusion Customer Relationship Management Domain 7-2
7.3 Adding a New Machine in the Oracle WebLogic Server Console 7-3
7.4 Packing and Unpacking the Managed Server Domain Home 7-3
7.5 Cloning Managed Servers and Assigning Them to CRMHOST2 7-4
7.6 Configuring Data Quality for Scale Out 7-6
7.6.1 Obtaining Postal Reference Data and License Keys 7-7
7.6.2 Setting Up the Data Quality Engine 7-7
7.6.3 Configuring the Data Quality Connector and IIR 7-8
7.6.4 Creating a Second Data Quality Server on CRMHOST2 7-9
7.7 Oracle HTTP Server Configuration 7-12
7.8 Validating the System 7-12

8 Scaling Out the Oracle Fusion Common Domain

8.1 Overview of the Oracle Fusion Common Domain 8-1
8.2 Prerequisites for Scaling Out the Oracle Fusion Common Domain 8-2
8.3 Adding a New Machine in the Oracle WebLogic Server Console 8-2
8.4 Scaling Out Oracle Universal Content Management 8-3
8.4.1 Creating a Common Location for the Oracle UCM Managed Servers 8-3
8.4.2 Scaling Out the Oracle UCM Inbound Refinery Server 8-3
8.5 Packing and Unpacking the Managed Server Domain Home 8-5
8.6 Cloning Managed Servers and Assigning Them to CRMHOST2 8-5
8.7 Oracle HTTP Server Configuration 8-8
8.8 Validating the System 8-9

9 Scaling Out the Oracle Fusion Human Capital Management Domain

9.1 Overview of the Oracle Fusion Human Capital Management Domain 9-1
9.2 Prerequisites for Scaling Out the Oracle Fusion Human Capital Management Domain. 9-2
9.3 Adding a New Machine in the Oracle WebLogic Server Console 9-3
9.4 Packing and Unpacking the Managed Server Domain Home 9-3
9.5 Cloning Managed Servers and Assigning Them to CRMHOST2 9-4
9.6 Oracle HTTP Server Configuration 9-6
9.7 Validating the System 9-7

10 Scaling Out the Oracle Fusion Supply Chain Management Domain

10.1 Overview of the Oracle Fusion Supply Chain Management Domain 10-1
10.2 Prerequisites for Scaling Out the Oracle Fusion Supply Chain Management Domain. 10-2
10.3 Adding a New Machine in the Oracle WebLogic Server Console 10-3
10.4 Packing and Unpacking the Managed Server Domain Home 10-3
10.5 Cloning Managed Servers and Assigning Them to CRMHOST2 10-4
10.6 Oracle HTTP Server Configuration 10-6
10.7 Validating the System 10-7

11 Scaling Out the Oracle Fusion Financials Domain

11.1 Overview of the Oracle Fusion Financials Domain 11-1
11.2 Prerequisites for Scaling Out the Oracle Fusion Financials Domain 11-2
11.3 Adding a New Machine in the Oracle WebLogic Server Console 11-3
11.4 Packing and Unpacking the Managed Server Domain Home 11-3
11.5 Cloning Managed Servers and Assigning Them to CRMHOST2 11-4
11.6 Oracle HTTP Server Configuration 11-6
11.7 Validating the System 11-7

12 Scaling Out the Oracle Fusion Incentive Compensation Domain

12.1 Overview of the Oracle Fusion Incentive Compensation Domain 12-1
12.2 Prerequisites for Scaling Out the Oracle Fusion Incentive Compensation Domain 12-2
12.3 Adding a New Machine in the Oracle WebLogic Server Console 12-3
12.4 Packing and Unpacking the Managed Server Domain Home 12-3
12.5 Cloning Managed Servers and Assigning Them to CRMHOST2 12-4
12.6 Oracle HTTP Server Configuration 12-6
12.7 Validating the System 12-7

13 Scaling Out the Oracle Business Intelligence Domain

13.1 Overview of the Oracle Business Intelligence Domain 13-1
13.2 Prerequisites for Scaling Out the Oracle Business Intelligence Domain 13-2
13.3 Starting the Default Node Manager 13-3
13.4 Prerequisites for Scaling Out Oracle Business Intelligence on CRMHOST2 13-3
13.4.1 Configuring JMS for Oracle BI Publisher 13-4
13.4.2 Setting the Listen Address for bi_server1 Managed Server 13-4
13.4.3 Updating the FusionVirtualHost_bi.conf Configuration File 13-5
13.5 Scaling Out Oracle Business Intelligence Components 13-6
13.5.1 Scaling Out the BI System on CRMHOST2 13-7
13.5.2 Start the Node Manager in SSL Mode 13-8
13.5.3 Scaling Out the System Components 13-9
13.5.4 Configuring Secondary Instances of Singleton System Components 13-9
13.5.5 Configuring the bi_server2 Managed Server 13-10
13.5.5.1 Setting the Listen Address for the bi_server2 Managed Server 13-10
13.5.5.2 Configuring Custom Identity and Custom Trust for the bi_server2 Managed Server 13-10
13.5.5.3 Disabling Host Name Verification for the bi_server2 Managed Server 13-12

13.5.6 Performing Additional Configuration for Oracle Business Intelligence High Availability 13-12
13.5.6.1 Additional Configuration Tasks for Oracle BI Scheduler 13-13
13.5.6.2 Additional Configuration Tasks for Oracle Real-Time Decisions 13-14
13.5.6.3 Additional Configuration Tasks for Oracle BI Publisher 13-15
13.5.6.4 Additional Configuration Tasks for Oracle BI for Microsoft Office 13-17
13.5.6.5 Additional Configuration Tasks for Oracle Financial Reporting 13-21
13.5.7 Configuring a Default Persistence Store for Transaction Recovery 13-21
13.5.8 Starting and Validating Oracle Business Intelligence on CRMHOST2 13-22
13.5.8.1 Starting the bi_server2 Managed Server 13-23
13.5.8.2 Starting the Oracle Business Intelligence System Components 13-23
13.5.8.3 Validating Oracle Business Intelligence URLs 13-23
13.5.9 Validating Access Through Oracle HTTP Server 13-24
13.5.10 Configuring Node Manager for the Managed Servers 13-24
13.5.11 Configuring Server Migration for the Managed Servers 13-25
13.6 Configuring Oracle Essbase Clustering Using the Essbase Failover Automation Tool 13-25
13.6.1 Prerequisites 13-25
13.6.2 Running the Essbase Failover Automation Tool 13-29
13.7 Validating the System 13-33

14 Additional Configuration Procedures for Scaling Out Oracle SOA Suite Server

14.1 Enabling Virtual IPs on CRMHOST1 and CRMHOST2 14-2
14.2 Setting the Listen Address for soa_server1 14-2
14.3 Setting the Listen Address for soa_server2 14-3
14.4 Updating the FusionVirtualHost_crm.conf Configuration File 14-3
14.5 Configuring JMS for the Oracle SOA Suite Server 14-4
14.6 Configuring Oracle Coherence for Deploying Composites 14-5
14.7 Configuring a Default Persistence Store for Transaction Recovery 14-7
14.8 Disabling Host Name Verification for the soa_server*n* Managed Servers 14-8
14.9 Restarting Node Manager on CRMHOST1 14-9
14.10 Starting and Validating soa_server1 on CRMHOST1 14-9
14.11 Restarting Node Manager on CRMHOST2 14-9
14.12 Starting and Validating soa_server2 on CRMHOST2 14-10

15 Configuring Administration Server High Availability

15.1 Enabling Administration Server High Availability 15-1
15.1.1 Enabling Administrative Virtual Host on CRMHOST1 15-2
15.1.2 Adding a New Machine in the Oracle WebLogic Server Console 15-3
15.1.3 Enabling the Administration Server to Listen on the Virtual IP Address 15-3
15.2 Oracle HTTP Server Configuration 15-4
15.3 Validating the Administration Server 15-5
15.4 Manually Failing Over the Administration Server to CRMHOST2 15-5
15.4.1 Prerequisites 15-5
15.4.2 Performing the Failover 15-5

15.5 Failing the Administration Server Back to CRMHOST1 15-6

16 Configuring Server Migration

16.1 Prerequisite 16-1
16.2 Migrating Oracle Fusion Applications Domains 16-1
16.2.1 About Configuring Server Migration 16-1
16.2.2 Setting Up a User and Tablespace for the Server Migration Leasing Table 16-2
16.2.3 Creating a Multi-Data Source Using the Oracle WebLogic Server Administration Console 16-3
16.2.4 Editing Node Manager's Properties File 16-4
16.2.5 Setting Environment and Superuser Privileges for the wlsifconfig.sh Script 16-5
16.2.6 Configuring Server Migration Targets 16-6
16.2.7 Testing the Server Migration 16-7

17 Configuring Oracle Business Intelligence Applications

17.1 Introduction to Oracle BI Applications for Oracle Fusion Customer Relationship Management 17-1
17.1.1 Topology 17-2
17.2 Preparing for an Oracle BI Applications Installation 17-3
17.2.1 Creating Databases for Oracle Business Intelligence Applications Components 17-4
17.2.2 Running Oracle BI Applications RCU to Create the Oracle BI Applications Schemas for the Data Warehouse 17-4
17.2.3 Installing and Configuring Informatica PowerCenter Services 17-8
17.2.4 Extending the Oracle Business Intelligence Domain by Deploying Oracle BI Applications Configuration Manager, Functional Setup Manager, and DAC 17-8
17.2.4.1 How to Configure DAC, Oracle BI Applications Configuration Manager, and Functional Setup Manager 17-8
17.2.4.2 Configuring Data Warehouse Administration Console for High Availability 17-11
17.3 Performing Additional Configuration Tasks 17-12
17.4 Configuring Oracle HTTP Server for the Managed Server 17-12
17.5 Performing Additional Data Warehouse Administration Console Tasks 17-13
17.6 Validating Oracle BI Applications Components URLs 17-13

18 Managing the Topology

18.1 Scaling the Topology for Additional Nodes 18-1
18.1.1 Scaling Out the Topology (Adding Managed Servers to a New Node) for Oracle ADF Server 18-2
18.1.1.1 Prerequisites for Scaling Out the Topology for Oracle ADF Server 18-2
18.1.1.2 Adding a New Machine in the Oracle WebLogic Server Console 18-2
18.1.1.3 Packing and Unpacking the Managed Server Domain Home 18-3
18.1.1.4 Cloning Managed Servers and Assigning Them to CRMHOST3 18-3
18.1.1.5 Validating the System 18-6
18.1.2 Scaling Up the Topology (Adding Managed Servers to an Existing Node) for Oracle ADF Server 18-7
18.1.2.1 Cloning Managed Servers and Assigning Them to CRMHOST3 18-7
18.1.2.2 Validating the System 18-8

18.1.3 Scaling Out the Topology (Adding Managed Servers to a New Node) for Oracle SOA Suite Server 18-9
18.1.3.1 Prerequisites for Scaling Out the Topology for Oracle SOA Suite Server 18-9
18.1.3.2 Adding a New Machine in the Oracle WebLogic Server Console 18-9
18.1.3.3 Packing and Unpacking the Managed Server Domain Home 18-10
18.1.3.4 Cloning Managed Servers and Assigning Them to CRMHOST3 18-10
18.1.3.5 Validating the System 18-13
18.1.3.6 Additional Configuration Procedures for Scaling Out Oracle SOA Suite Server 18-14
18.1.4 Scaling Up the Topology (Adding Managed Servers to an Existing Node) for Oracle SOA Suite Server 18-19
18.1.4.1 Cloning Managed Servers and Assigning Them to CRMHOST3 18-19
18.1.4.2 Validating the System 18-20
18.1.5 Scaling Out the Topology (Adding Managed Servers to a New Node) for Oracle Business Intelligence 18-20
18.1.5.1 Prerequisites for Scaling Out the Topology for Oracle Business Intelligence 18-21
18.1.5.2 Scale-Out Procedure for Oracle Business Intelligence 18-21
18.1.6 Scaling Up the Topology for Oracle Business Intelligence 18-23
18.1.6.1 Scale-Up Procedure for Oracle Business Intelligence 18-23
18.2 Performing Backups and Recoveries 18-24
18.3 Monitoring the Topology 18-24
18.4 Migrating from a Test Environment to a Production Environment 18-25
18.5 Configuring Log File Rotation 18-25
18.5.1 Specifying Log File Rotation Using Fusion Middleware Control 18-25
18.5.2 Specifying Log File Rotation Using WLST 18-26
18.6 Patching the Topology 18-26
18.7 Auditing 18-26
18.8 Troubleshooting 18-28
18.8.1 Page Not Found When Accessing soa-infra Application Through Load Balancer 18-29
18.8.2 Soa-infra Application Fails to Start Due to Deployment Framework Issues (Coherence) 18-30
18.8.3 Incomplete Policy Migration After Failed Restart of SOA Server 18-30
18.8.4 Oracle SOA Suite Server Fails to Start Due to Maximum Number of Processes Available in Database 18-31
18.8.5 Administration Server Fails to Start After a Manual Failover 18-31
18.8.6 Error While Activating Changes in Administration Console 18-32
18.8.7 SOA Server Not Failed Over After Server Migration 18-32
18.8.8 SOA Server Not Reachable From Browser After Server Migration 18-32
18.8.9 Oracle Access Manager Configuration Tool Does Not Remove URLs 18-33
18.8.10 Redirecting of Users to Login Screen After Activating Changes in Administration Console 18-33
18.8.11 Redirecting of Users to Administration Console's Home Page After Activating Changes to OAM 18-33
18.8.12 Configured JOC Port Already in Use 18-33
18.8.13 Out-of-Memory Issues on Managed Servers 18-34
18.8.14 JDBC Connection Reset Appears When on OEL 5.4 18-34
18.8.15 Missing JMS Instances on Oracle BI Publisher Scheduler Diagnostics Page 18-34

18.8.16 Oracle BI Publisher Jobs in Inconsistent State After Managed Server Shutdown 18-35
18.8.17 JMS Instance Fails in an Oracle BI Publisher Cluster 18-35
18.8.18 Administration Console Redirects from Internal URL to Container URL after Activation..... 18-35

A Protected Domain URIs

A.1 Protected CRM Domain URIs A-1
A.2 Protected BI Domain URIs..... A-3

Preface

This preface describes the audience, contents and conventions used in the *Oracle Fusion Applications Enterprise Deployment Guide*

Audience

This guide is intended for system administrators who are responsible for implementing and configuring Oracle Fusion Applications 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

For more information, see the following documents in the Oracle Fusion Applications documentation set or in the Oracle Fusion Middleware documentation set:

- *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence*
- *Oracle Fusion Middleware Configuration Guide for Oracle Business Intelligence Applications*
- *Oracle Fusion Applications Administrator's Guide*
- *Oracle Database Administrator's Guide*
- *Oracle Fusion Applications Installation Guide*
- *Oracle Fusion Middleware Repository Creation Utility User's Guide*
- *Oracle Real Application Clusters Administration and Deployment Guide*
- *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management*
- *Oracle Fusion Applications Concepts Guide*

Conventions

The following text conventions are used in this document:

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

What's New in This Guide for Release 11.1.1.5 (September)

This chapter provides information about what has been changed in or added to the *Oracle Fusion Applications Enterprise Deployment Guide*, 11*g* since Release 1 (11.1.1.5) of August 2011.

Documentation Changes for Release 11.1.1.5 (September)

For this release, all chapters in this guide have been updated to reflect current enterprise-deployment requirements, procedures, and specifications. The following table lists new sections that have been added and those that have undergone major revisions since the last release.

For changes made to Oracle JDeveloper and Oracle Application Development Framework (Oracle ADF) for this release, see the What's New page on the Oracle Technology Network at `http://www.oracle.com/technetwork/developer-tools/jdev/documentation/index.html`.

Sections	Changes Made
Chapter 2 Network Configuration	
Section 2.6, "IPs and Virtual IPs"	Table 2–1 has been updated to include new VIPs.
Chapter 3 Setting Up the Database Tier	
Section 3.3.1, "Updating the Kernel Parameters"	Added new section desribing how to update the kernel parameters for Linux before installing the database.
Section 3.4, "Loading the Oracle Fusion Applications Repository into the Oracle RAC Database"	Section has been updated with current images and descriptions.
Chapter 4 Using the Provisioning Process to Install Components for an Enterprise Deployment	
Section 4.3.2, "Creating a New Provisioning Plan"	Section has been updated with current images and descriptions. Some images have been removed.
Chapter 13 Scaling Out the Oracle Business Intelligence Domain	
Section 13.6, "Configuring Oracle Essbase Clustering Using the Essbase Failover Automation Tool"	Added prerequisite to update the `arborPATH` property.
Chapter 17 Configuring Oracle Business Intelligence Applications	

Sections	Changes Made
Section 17.5, "Performing Additional Data Warehouse Administration Console Tasks"	Added new section describing additional tasks.

1 Enterprise Deployment Overview

This chapter provides an overview of the enterprise topology for Oracle Fusion Customer Relationship Management.

This chapter includes the following topics:

- Section 1.1, "What is an Enterprise Deployment?"
- Section 1.2, "About Oracle Fusion Applications"
- Section 1.3, "Benefits of Oracle Recommendations"
- Section 1.4, "Terminology"
- Section 1.5, "Reference Enterprise Deployment Topology"
- Section 1.6, "Hardware Requirements"
- Section 1.7, "Installation Prerequisite"
- Section 1.8, "Implementing the Topology"

1.1 What is an Enterprise Deployment?

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

An Oracle Fusion Applications enterprise deployment:

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

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

1.2 About Oracle Fusion Applications

Oracle Fusion Applications are a unified suite of business applications designed to unify personal and enterprise processes. It unifies transactional Oracle SOA Suite and business processes, business intelligence, and collaborative technologies in a seamless user experience. Oracle Fusion Applications can be easily integrated into a service-oriented architecture and made available as software as a service.

Oracle Fusion Applications offer a strong functional value by providing:

- Installed based demand (for example, unified global payroll module)
- Competitive differentiation (for example, Distributed Order Orchestration)
- Revenue generation (for example, sales territory management)

Oracle Fusion Applications incorporate best practice business processes, including those from Oracle E-Business Suite, PeopleSoft, Oracle On Demand, JD Edwards, and Siebel.

Oracle Fusion Applications are standards-based, making them highly adaptable. This standards-based technology allows you to respond effectively to change with flexible, modular, user-driven business software that is powered by best-in-class business capabilities built on open standards. Its technology framework includes the following products:

- Oracle WebCenter provides design time and runtime tools for building enterprise portals, transactional websites, and social networking sites.
- Oracle Business Intelligence 11*g* provides a full range of business intelligence capabilities that allow you to collect, present, and deliver organizational data.
- Hyperion extends Oracle's business intelligence capabilities to offer the most comprehensive system for enterprise performance management.
- Oracle Universal Content Management enables you to leverage document management, Web content management, digital asset management, and records retention functionality to build and complement your business applications.
- Oracle SOA Suite provides an enterprise architecture that supports building connected enterprise applications to provide solutions to business problems.
- Oracle WebLogic Server is a scalable, enterprise-ready Java Platform, Enterprise Edition (Java EE) application server.
- Oracle JDeveloper is an integrated development environment with end-to-end support for modeling, developing, debugging, optimizing, and deploying Java applications and Web services.
- Oracle Enterprise Manager Fusion Middleware Control offers business-driven applications management, integrated application to disk management, and integrated systems management and support experience.
- Oracle Identity Management enables organizations to manage the end-to-end life cycle of user identities and to secure access to enterprise resources and assets.

For more information, see the *Oracle Fusion Applications Concepts Guide*.

1.3 Benefits of Oracle Recommendations

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

- Built-in Security
- High Availability

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

1.3.1 Built-in Security

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

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

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

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

1.3.2 High Availability

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

1.4 Terminology

The following terminology is used in this enterprise deployment guide:

- **Oracle home**: An Oracle home contains installed files necessary to host a specific product. For example, the SOA Oracle home contains a directory that contains binary and library files for Oracle SOA Suite. An Oracle home resides within the directory structure of the Middleware home. Each Oracle home can be associated with multiple Oracle instances or Oracle WebLogic Server domains.
- ***ORACLE_BASE***: An alternate way of specifying the path `/u01/oracle`.
- **WebLogic Server home**: A WebLogic Server home contains installed files necessary to host a WebLogic Server. The WebLogic Server home directory is a peer of Oracle home directories and resides within the directory structure of the Middleware home.
- **Middleware home:** A Middleware home consists of the Oracle WebLogic Server home, and, optionally, one or more Oracle homes. A Middleware home can reside on a local file system or on a remote shared disk that is accessible through NFS.
- **Oracle instance**: An Oracle instance contains one or more active middleware system components, for example Oracle Web Cache, Oracle HTTP Server, or Oracle Internet Directory. You determine which components are part of an instance, either at install time or by creating and configuring an instance at a later time. An Oracle instance contains files that can be updated, such as configuration files, log files, temporary files.
- **Domain**: The basic administrative unit of Oracle WebLogic Server.
- **Managed Server**: Hosts business applications, application components, Web services, and their associated resources. To optimize performance, Managed Servers maintain a read-only copy of the domain's configuration document. When a Managed Server starts, it connects to the domain's Administration Server to synchronize its configuration document with the document that the Administration Server maintains.
- **failover**: When a member of a high availability system fails unexpectedly (unplanned downtime), in order to continue offering services to its consumers, the system undergoes a failover operation. If the system is an active-passive system, the passive member is activated during the failover operation and consumers are directed to it instead of the failed member. The failover process can be performed manually, or it can be automated by setting up hardware cluster services to detect failures and move cluster resources from the failed node to the standby node. If the system is an active-active system, the failover is performed by the load balancer entity serving requests to the active members. If an active member fails, the load balancer detects the failure and automatically redirects requests for the failed member to the surviving active members. See *Oracle Fusion Middleware High Availability Guide* for information on active-active and active-passive systems.
- **failback**: After a system undergoes a successful failover operation, the original failed member can be repaired over time and be re-introduced into the system as a standby member. If desired, a failback process can be initiated to activate this member and deactivate the other. This process reverts the system back to its pre-failure configuration.
- **hardware cluster**: A hardware cluster is a collection of computers that provides a single view of network services (for example: an IP address) or application services (for example: databases, Web servers) to clients of these services. Each node in a hardware cluster is a standalone server that runs its own processes.

These processes can communicate with one another to form what looks like a single system that cooperatively provides applications, system resources, and data to users.

A hardware cluster achieves high availability and scalability through the use of specialized hardware (cluster interconnect, shared storage) and software (health monitors, resource monitors). The cluster interconnect is a private link used by the hardware cluster for heartbeat information to detect node death. Due to the need for specialized hardware and software, hardware clusters are commonly provided by hardware vendors such as 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 Applications high availability, only two nodes are required. Hence, this document assumes a two-node hardware cluster for high availability solutions employing a hardware cluster.

- **cluster agent**: The software that runs on a node member of a hardware cluster that coordinates availability and performance operations with other nodes. Clusterware provides resource grouping, monitoring, and the ability to move services. A cluster agent can automate the service failover.
- **clusterware**: Software that manages the operations of the members of a cluster as a system. It allows one to define a set of resources and services to monitor via a heartbeat mechanism between cluster members and to move these resources and services to a different member in the cluster as efficiently and transparently as possible.
- **shared storage**: Shared storage is the storage subsystem that is accessible by all the computers in the enterprise deployment domain. Among other things, the following is located on the shared disk:
  - Middleware Home software
  - AdminServer Domain Home
  - Java Message Service (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 Applications instance at any given time and has been configured to have a backup/secondary node. If the primary node fails, Oracle Fusion Applications instance is failed over to the secondary node. This failover can be manual or automated using the Clusterware for Administration Server. For a server migration based scenario, WebLogic Whole Server Migration is used for automated failover.
- **secondary node**: The node that is the backup node for an Oracle Fusion Applications instance. This is where the active instance fails over when the primary node is no longer available. See the definition for primary node in this section.
- **network host name**: Network host name is a name assigned to an IP address either through the `/etc/hosts` file or through DNS resolution. This name is visible in the network that the computer to which it refers to is connected. Often, the network host name and physical host name are identical. However, each computer has only one physical host name but may have multiple network host

names. Thus, a computer's network host name may not always be its physical host name.

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

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

- **physical IP**: Physical IP refers to 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 of the physical host name). In contrast to a virtual IP, it is always associated with the same computer when on a network.

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

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

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

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

  A hardware cluster uses a cluster virtual IP to present to the outside world the entry point into the cluster (it can also be set up on a standalone 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 could be several cluster IP

addresses. These cluster IP addresses float or migrate between the two nodes. The node with current ownership of a cluster IP address is active for that address.

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

1.5 Reference Enterprise Deployment Topology

The instructions and diagrams in this guide describe a reference enterprise deployment topology for Oracle Fusion Applications:

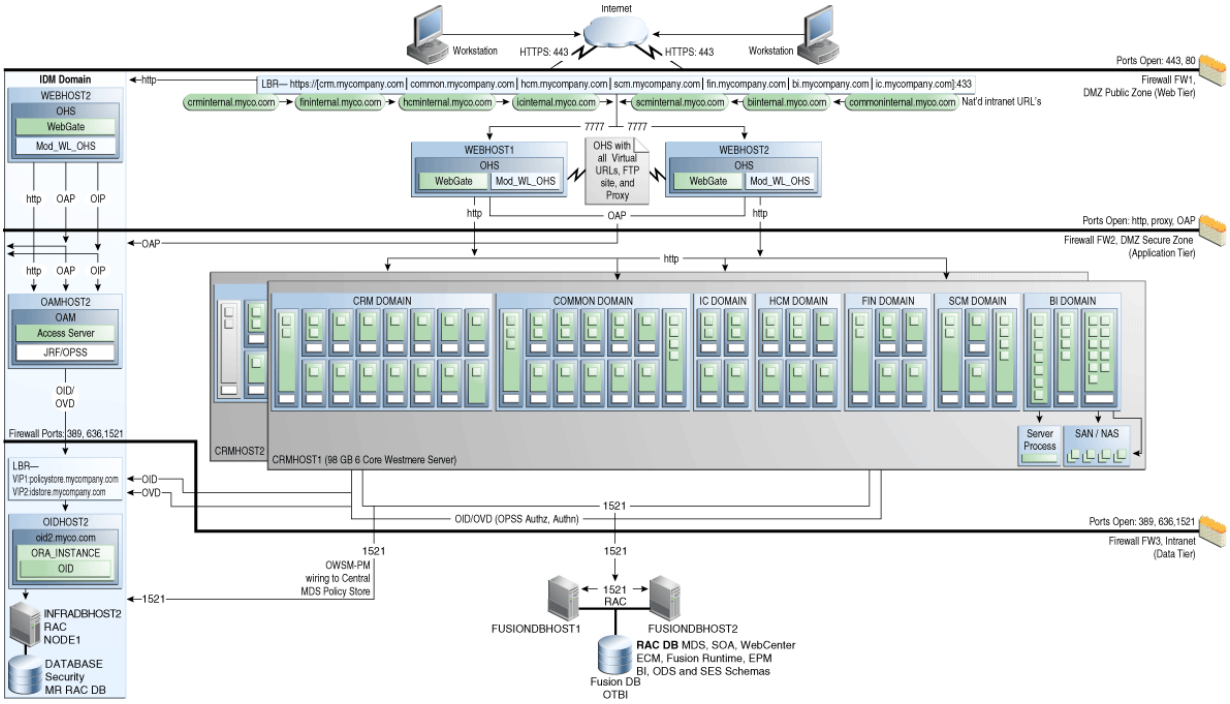
- Overall Reference Enterprise Deployment Topology
- Oracle Web Tier
- Application Tier
- Data Tier

1.5.1 Overall Reference Enterprise Deployment Topology

Figure 1–1 shows the overall Oracle Fusion Customer Relationship Management reference enterprise deployment topology, to which variations may be applied. The graphic illustrates how all components are deployed together.

In the topology, the primary node (also known as a host, and is *CRMHOST1* in the diagram) is actively running the Oracle Fusion Applications instance. The secondary node (*CRMHOST2*) is the redundant (HA) node for the Oracle Fusion Applications instance. The primary node consists of an Administration Server and applications that have been deployed to Managed Servers. Managed Servers can be grouped together in clusters to provide scalability and high availability for applications. Together, the primary and secondary nodes form a domain.

Figure 1–1 Oracle Fusion CRM Reference Enterprise Deployment Topology

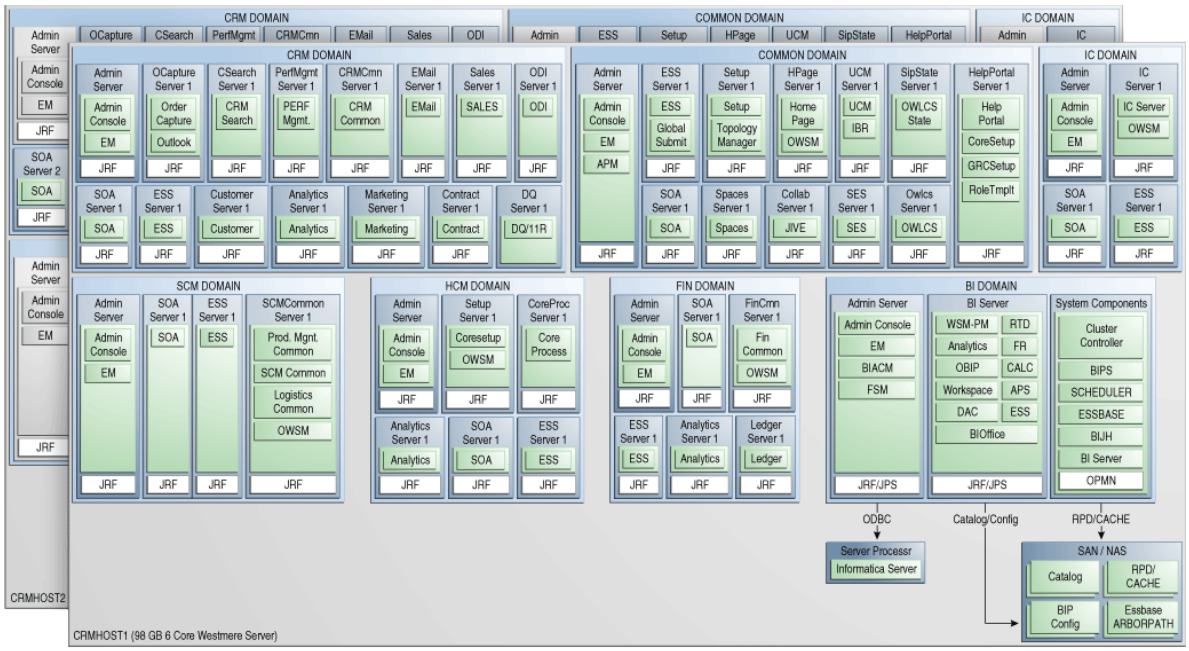


As shown in Figure 1–1, the overall Oracle Fusion Applications reference enterprise deployment topology comprises several domains:

- Oracle Fusion Customer Relationship Management Domain
- Oracle Fusion Setup Domain
- Oracle Fusion Financials Domain
- Oracle Fusion Human Capital Management Domain
- Oracle Fusion Supply Chain Management Domain
- Oracle Fusion Incentive Compensation Domain
- Oracle Business Intelligence Domain

Figure 1–2 shows each of the domains in detail.

Figure 1–2 Domain Details



The scale out of these domains is described in the chapters that follow.

For information about installing the Oracle Identity Management stack for Oracle Fusion Applications, see Section 4.2, "Prerequisites for Using the Provisioning Process."

1.5.2 Oracle Web Tier

Nodes in the Oracle Web Tier are located in the demilitarized zone (DMZ) public zone. In this tier, two nodes *WEBHOST1* and *WEBHOST2* run Oracle HTTP Server configured with WebGate and FusionVirtualHost_*domain*.conf.

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

WebGate (which is an Oracle Access Manager component) in Oracle HTTP Server uses Oracle Access Protocol (OAP) to communicate with Oracle Access Manager running on *OAMHOST1* and *OAMHOST2*, in the Identity Management DMZ. WebGate and Oracle Access Manager are used to perform operations such as user authentication.

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

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

On the firewall protecting the Oracle Web Tier, only the HTTP ports are open: 443 for HTTPS and 80 for HTTP.

Load Balancer Requirements

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

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

1.5.3 Application Tier

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

CRMHOST1 and *CRMHOST2* run all the managed servers in the Oracle Fusion Customer Relationship Management, Oracle Business Intelligence, Oracle Incentive Compensation, Oracle Fusion Financials, Oracle Fusion Supply Chain Management, and Oracle Fusion Human Capital Management domains.

CRMHOST1 and *CRMHOST2* run the managed and C/C++ servers from different domains in an active-active or active-passive implementation. C/C++ components are managed by Oracle Process Manager and Notification Server (OPMN), and all the managed servers are managed by Administration Server within the domain.

CRMHOST1 and *CRMHOST2* also run the Oracle WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control, but in an active-passive configuration. You also can fail over the Administration Server

manually. Alternatively, you can configure the Oracle WebLogic Server Administration Console with CFC/CRS to fail over automatically on a separate hardware cluster.

Oracle Web Services Manager (Oracle WSM) provides a policy framework to manage and secure Web services in the enterprise deployment topology. WSM Policy Manager also runs in active-active configuration in every Fusion domain where Web Services are hosted.

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

1.5.4 Data Tier

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

In this tier, an Oracle RAC database runs on the nodes *FUSIONDBHOST1* and *FUSIONDBHOST2*. The database contains the schemas needed by the Oracle Fusion Applications components. The components running in the application tier access this database.

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

1.6 Hardware Requirements

This section provides recommended hardware for the Oracle Fusion Applications reference enterprise deployment topology on Linux operating systems.

The recommended hardware for the Oracle Fusion Applications reference enterprise deployment topology consists of six 96 GB Intel Westmere, six dual-core CPU servers (excluding Oracle HTTP Server and Oracle Database servers). Table 1–1 describes the typical hardware requirements.

Table 1–1 Typical Hardware Requirements

Server	Processor	Memory	TMP	SWAP
CRMHOST1	6 core 2 CPU Westmere	96 GB	default	default
CRMHOST2	6 core 2 CPU Westmere	96 GB	default	default
WEBHOST1	2 core 2 CPU	4 GB	default	default
WEBHOST2	2 core 2 CPU	4 GB	default	default
FUSIONDBHOST1	4 core 4 CPU	16 GB	default	default
FUSIONDBHOST2	4 core 4 CPU	16 GB	default	default

1.7 Installation Prerequisite

The Oracle Identity Management stack for Oracle Fusion Applications should already be installed prior to starting a deployment. Note, however, that the provisioning process described in Chapter 4, "Using the Provisioning Process to Install Components for an Enterprise Deployment," cannot proceed without it. Follow the instructions in *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management* to install and configure these components.

1.8 Implementing the Topology

Oracle recommends the following approach when implementing the Oracle Fusion Applications topology outlined in Section 1.5, "Reference Enterprise Deployment Topology":

1. Network Configuration
2. Setting Up the Database Tier
3. Using the Provisioning Process to Install Components for an Enterprise Deployment
4. Scaling Out Oracle HTTP Server
5. Configuring Node Manager
6. Scaling Out the Oracle Fusion Customer Relationship Management Domain
7. Scaling Out the Oracle Fusion Common Domain
8. Scaling Out the Oracle Fusion Human Capital Management Domain
9. Scaling Out the Oracle Fusion Supply Chain Management Domain
10. Scaling Out the Oracle Fusion Financials Domain
11. Scaling Out the Oracle Fusion Incentive Compensation Domain
12. Scaling Out the Oracle Business Intelligence Domain
13. Additional Configuration Procedures for Scaling Out Oracle SOA Suite Server
14. Configuring Administration Server High Availability
15. Configuring Server Migration
16. Configuring Oracle Business Intelligence Applications
17. Managing the Topology

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

2

Network Configuration

This chapter describes the network environment configuration required by the Oracle Fusion Applications reference enterprise deployment topology, as well as recommendations for shared storage and directory structure. It contains the following topics:

- Section 2.1, "External Virtual Server Name"
- Section 2.2, "Internal Virtual IP"
- Section 2.3, "Load Balancer Configuration"
- Section 2.4, "Reference Enterprise Deployment Directory Structure"
- Section 2.5, "Shared Storage"
- Section 2.6, "IPs and Virtual IPs"
- Section 2.7, "Firewalls and Ports"
- Section 2.8, "Clock Synchronization"

2.1 External Virtual Server Name

The Oracle Fusion Applications enterprise deployment topology uses the following externally accessible load balancer virtual IPs that are created on the Load Balancer.

- *crmexternal*.mycompany.com
- *finexternal*.mycompany.com
- *hcmexternal*.mycompany.com
- *scmexternal*.mycompany.com
- *biexternal*.mycompany.com
- *commonexternal*.mycompany.com
- *icexternal*.mycompany.com

These virtual server names act as the access point for all HTTP traffic to the runtime components for Oracle Fusion Customer Relationship Management. The HTTP traffic from client browser to LBR is always in SSL.

These VIPs receive all the requests externally (from the intranet or internet) on port 443 in SSL mode. These requests are forwarded to one of Oracle HTTP Server's "external virtual hosts specific to each domain" on `WEBHOST1` or `WEBHOST2`.

Note: All external VIPs listed above are also configured on port 80, and any request that is received on port 80 will be forwarded back to port 443. This is to prevent a browser error when the user types the URL without the `http://` and the browser uses the default 80 port. If the user types `https://`, the browser uses the default 443 port .

2.2 Internal Virtual IP

The following Oracle Fusion Customer Relationship Management deployment topology also requires separate secure network address translations (NATs) internal VIPS for each domain. These VIPs are used for transactional and administrative access.

- *crminternal*.mycompany.com
- *fininternal*.mycompany.com
- *hcminternal*.mycompany.com
- *scminternal*.mycompany.com
- *biinternal*.mycompany.com
- *commoninternal*.mycompany.com
- *icinternal*.mycompany.com

The above virtual URLs (VIPs) are defined on the load balancer and are used for internal invocations of services within the data center. The URLs are not exposed to the internet or intranet, and are only accessible within the data center.

The VIPs receive all the requests internally on port 7777 in non-SSL mode. All the internal services/clients access these VIPs using the above virtual addresses, and the requests are then forwarded to one of Oracle HTTP Server's "internal virtual hosts specific to each domain" on *`WEBHOST1`* or *`WEBHOST2`*.

For additional Oracle WebLogic Server security, you can configure the internal VIPs listed above with a load-balancing router rule that accepts requests only from well-known hosts like *`CRMHOST1`*, *`CRMHOST2`*, or from the system administrator host, and rejects all other requests.

2.3 Load Balancer Configuration

The Oracle Fusion Applications enterprise topology requires an external load balancer with SSL acceleration. To configure the load balancer with above VIPs listed above, refer to vendor-specific load balancer configuration instructions.

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

2.4 Reference Enterprise Deployment Directory Structure

This section describes the directory structure specifically used by the Oracle Fusion Applications reference enterprise deployment topology. It includes the following topics:

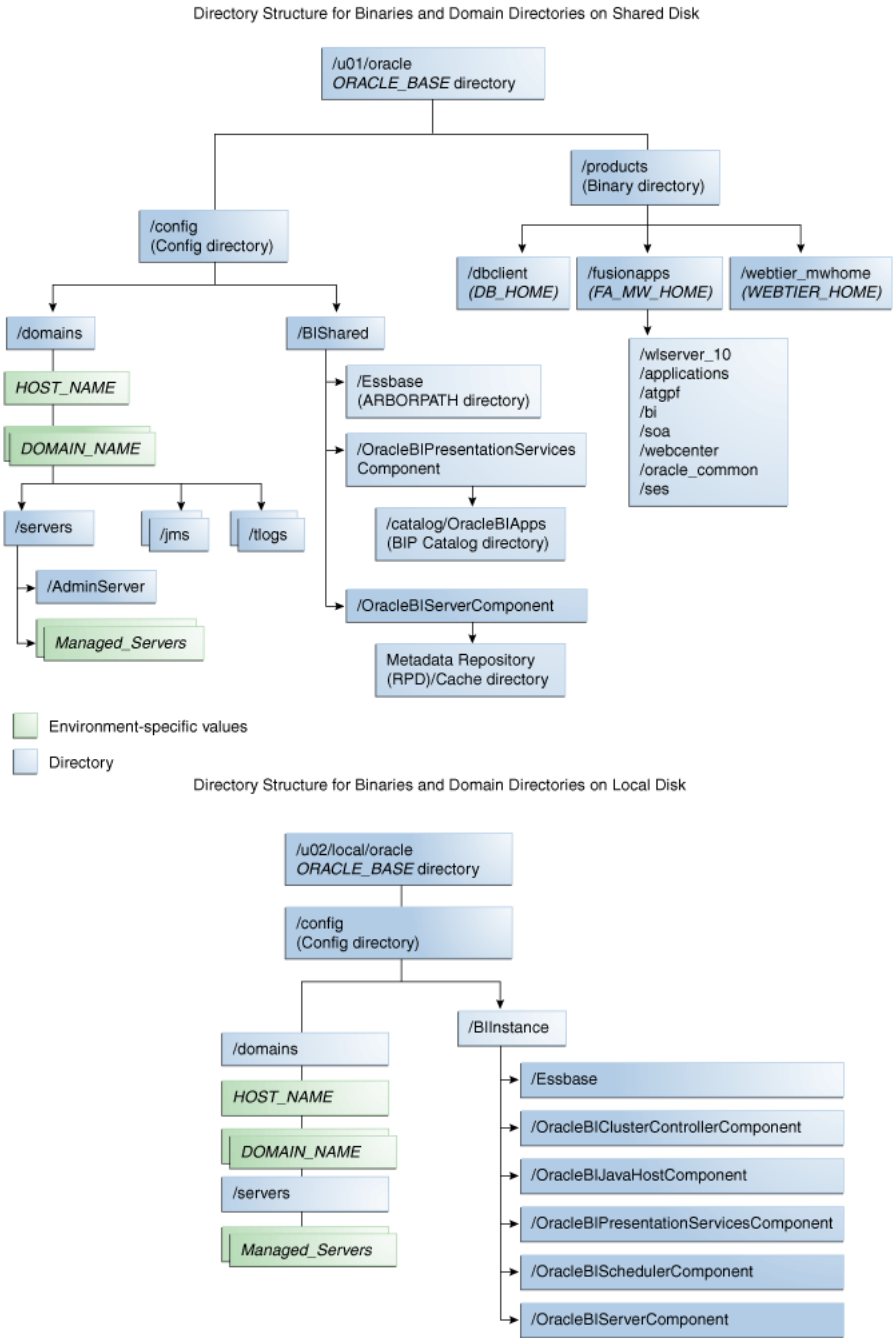
- Section 2.4.1, "Directory Structure"
- Section 2.4.2, "Binary Directory Structure"
- Section 2.4.3, "Domain Configuration Directory Structure"

For general information about Oracle Fusion Applications architecture and concepts, see "Introduction to Oracle Fusion Applications for System Administrators" in *Oracle Fusion Applications Administrator's Guide*.

2.4.1 Directory Structure

Figure 2–1 shows the enterprise deployment directory structure and its dependencies.

Figure 2–1 Enterprise Deployment Directory Structure for Oracle Fusion Applications



2.4.2 Binary Directory Structure

The binaries in the Oracle Fusion Applications reference enterprise deployment topology (the Oracle Fusion Middleware home and Oracle home) are on a shared disk. In order to avoid disk corruption, you may choose to maintain snapshots.

The file system for the binaries should be mounted on all the nodes with the exact mount point and path. For example, `/u01/oracle`.

2.4.3 Domain Configuration Directory Structure

The domain configuration directory structure is created on a shared disk, and its mount point should be visible from all nodes This path will be used by the components that require shared resources and also administration servers to start active-passive processes. For example, `/u01/oracle`.

2.5 Shared Storage

For binaries: the file system is optimized for read operations.

For config: the file system should be optimized for read/write operations. For example, AdminServer Domain directory and Oracle Business Intelligence shared folders like Oracle Business Intelligence WebCat, RPD cache, Essbase ARBORPATH, and Oracle Business Intelligence config.

The following steps show how to create and mount shared storage locations for binaries and config so that *CRMHOST1* and *CRMHOST2* can see the same location.

"nasfiler" is the shared storage filer.

From *CRMHOST1*:

```
CRMHOST1> mount nasfiler:/vol/vol1/u01/oracle /u01/oracle -t nfs
```

From *CRMHOST2*:

```
CRMHOST2> mount nasfiler:/vol/vol1/u01/oracle /u01/oracle -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 *CRMHOST1*. The options may differ.

```
mount nasfiler:/vol/vol1/u01/oracle
/u01/oracle -t nfs -o
rw,bg,hard,nointr,tcp,vers=3,timeo=300,rsize=32768,
wsize=32768
```

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

2.5.1 Shared Storage for Oracle Business Intelligence

For general information, see "Shared Storage and Recommended Directory Structure" in *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence*.

Shared Storage Locations

In addition, Oracle Business Intelligence has two specific shared-storage locations.

- Location for Data Warehouse Console Configuration folder:

  ORACLE_BASE`/config/BIShared/dac`

  – Mounted from: All nodes containing the instance of DAC in the cluster or where DAC can be migrated to must mount this location (all nodes must have read/write access)

- Location for shared Essbase ARBORPATH:

  ORACLE_BASE`/config/BIShared/Essbase`

– Mounted from: All nodes containing the instance of Essbase in the cluster must mount this location (all nodes must have read/write access)

Note: *ORACLE_BASE* is `/u01/oracle`.

2.6 IPs and Virtual IPs

Configure the Administration Server and the managed servers to listen on different virtual IPs and physical IPs.

The following VIPs are required to configure specific components:

- Virtual IPs for AdminServer are needed for every domain to configure AdminServer in active-passive mode. These VIPs are shared across *CRMHOST1* and *CRMHOST2*, depending on where the AdminServer is running.
- Virtual IPs for all Oracle SOA Suite servers in every domain, and Oracle Business Intelligence servers in the Oracle Business Intelligence domain are needed to support server migration. These components are implemented in active-active mode, so these VIPs are needed for *CRMHOST1* and *CRMHOST2*.

Table 2–1 provides descriptions of the various virtual hosts.

Table 2–1 Virtual Hosts

Virtual IP	VIP Maps to...	Description
VIP1	*CRMADMINVH*	The virtual host name that is the listen address for the CRM Administration Server and fails over with manual failover of the Administration Server. It is enabled on the node where the CRM Administration Server process is running (*CRMHOST1* by default).
VIP2	*CRMSOAVH1*	The virtual host name that maps to the listen address for `soa_server1` and fails over with server migration of this managed server. It is enabled on the node where `soa_server1` process is running (*CRMHOST1* by default).
VIP3	*CRMSOAVH2*	The virtual host name that maps to the listen address for `soa_server2` and fails over with server migration of this managed server. It is enabled on the node where `soa_server2` process is running (*CRMHOST2* by default).
VIP4	*COMMONADMINVH*	The virtual host name that is the listen address for the CRM Administration Server and fails over with manual failover of the Administration Server. It is enabled on the node where the CRM Administration Server process is running (*CRMHOST1* by default).
VIP5	*COMMONSOAVH1*	The virtual host name that maps to the listen address for `soa_server1` and fails over with server migration of this managed server. It is enabled on the node where `soa_server1` process is running (*CRMHOST1* by default).
VIP6	*COMMONSOAVH2*	The virtual host name that maps to the listen address for `soa_server2` and fails over with server migration of this managed server. It is enabled on the node where `soa_server2` process is running (*CRMHOST2* by default).

Table 2–1 (Cont.) Virtual Hosts

Virtual IP	VIP Maps to...	Description
VIP7	*FINADMINVH*	The virtual host name that is the listen address for the CRM Administration Server and fails over with manual failover of the Administration Server. It is enabled on the node where the CRM Administration Server process is running (*CRMHOST1* by default).
VIP8	*FINSOAVH1*	The virtual host name that maps to the listen address for `soa_server1` and fails over with server migration of this managed server. It is enabled on the node where `soa_server1` process is running (*CRMHOST1* by default).
VIP9	*FINSOAVH2*	The virtual host name that maps to the listen address for `soa_server2` and fails over with server migration of this managed server. It is enabled on the node where `soa_server2` process is running (*CRMHOST2* by default).
VIP10	*HCMADMINVH*	The virtual host name that is the listen address for the CRM Administration Server and fails over with manual failover of the Administration Server. It is enabled on the node where the CRM Administration Server process is running (*CRMHOST1* by default).
VIP11	*HCMSOAVH1*	The virtual host name that maps to the listen address for `soa_server1` and fails over with server migration of this managed server. It is enabled on the node where `soa_server1` process is running (*CRMHOST1* by default).
VIP12	*HCMSOAVH2*	The virtual host name that maps to the listen address for `soa_server2` and fails over with server migration of this managed server. It is enabled on the node where `soa_server2` process is running (*CRMHOST2* by default).
VIP13	*SCMADMINVH*	The virtual host name that is the listen address for the CRM Administration Server and fails over with manual failover of the Administration Server. It is enabled on the node where the CRM Administration Server process is running (*CRMHOST1* by default).
VIP14	*SCMSOAVH1*	The virtual host name that maps to the listen address for `soa_server1` and fails over with server migration of this managed server. It is enabled on the node where `soa_server1` process is running (*CRMHOST1* by default).
VIP15	*SCMSOAVH2*	The virtual host name that maps to the listen address for `soa_server2` and fails over with server migration of this managed server. It is enabled on the node where `soa_server2` process is running (*CRMHOST2* by default).
VIP16	*BIADMINVH*	The virtual host name that is the listen address for the BI Domain Administration Server. It is enabled on the node where the BI Domain Administration Server process is running (*CRMHOST1* by default).
VIP17	*BIVH1*	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 `bi_server1` process is running (*CRMHOST1* by default).

Table 2–1 (Cont.) Virtual Hosts

Virtual IP	VIP Maps to...	Description
VIP18	*BIVH2*	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 bi_server2 process is running (*CRMHOST2* by default).
VIP19	*FUSIONDBHOST1*	The virtual host name that is the listen address for the database Oracle RAC database server. It is enabled on the node where the database is running.
VIP20	*CRMSOAVH3*	The virtual host name that maps to the listen address for soa_server3 and fails over with server migration of this managed server. It is enabled on the node where soa_server3 process is running (*CRMHOST3* by default).
VIP21	*ICSOAVH1*	The virtual host name that maps to the listen address for soa_server1 and fails over with server migration of this managed server. It is enabled on the node where soa_server1 process is running (*CRMHOST1* by default).
VIP22	*ICSOAVH2*	The virtual host name that maps to the listen address for soa_server2 and fails over with server migration of this managed server. It is enabled on the node where soa_server2 process is running (*CRMHOST2* by default).
VIP23	*ICADMINVH*	The virtual host name that is the listen address for the IC Domain Administration Server. It is enabled on the node where the IC Domain Administration Server process is running (*CRMHOST1* by default).

2.7 Firewalls and Ports

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

Most port numbers are assigned during installation.

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

Firewall notation:

- FW1 refers to the outermost firewall.
- FW2 refers to the firewall between the Oracle Web Tier and the application tier.
- FW3 refers to the firewall between the application tier and the data tier.

Table 2–2 Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Browser request	FW1	80	HTTP	Inbound	Timeout depends on all HTML content and the type of process model used for Oracle Fusion Applications.
Browser request	FW1	443	HTTPS / Load Balancer	Inbound	Timeout depends on all HTML content and the type of process model used for Oracle Fusion Applications.
OAP	n/a	8181	HTTP	n/a	
Oracle HTTP Server registration with Administration Server	FW2	7001	HTTP/t3	Inbound	Set the timeout to a short period (5-10 seconds).
Oracle HTTP Server registration with Administration Server	FW2	OPMN port (6701) and OHS Admin Port (7779)	TCP and HTTP, respectively	Outbound	Set the timeout to a short period (5-10 seconds).
Common Domain	FW2	7001-7035	HTTP	n/a	
Financial Domain	FW2	7401-7430	HTTP	n/a	
Supply Chain Domain	FW2	7801-7830	HTTP	n/a	
Customer Relationship Management Domain	FW2	9001-9040	HTTP	n/a	
Human Capital Management Domain	FW2	9401-9430	HTTP	n/a	
Business Intelligence Domain	FW2	10201-10230	HTTP	n/a	
Incentive Compensation Domain	FW2	9801-9830	HTTP	n/a	
Common Administration Console access	FW2	7001	HTTP / Administration Server and Enterprise Manager t3	Both	You should tune this timeout based on the type of access to the administration console (whether it is planned to use the Oracle WebLogic Server Administration Console from application tier clients or clients external to the application tier).
Financial Administration Console access	FW2	7401	HTTP / Administration Server and Enterprise Manager t3	Both	You should tune this timeout based on the type of access to the administration console (whether it is planned to use the Oracle WebLogic Server Administration Console from application tier clients or clients external to the application tier).

Table 2–2 (Cont.) Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
SCM Administration Console access	FW2	7801	HTTP / Administration Server and Enterprise Manager t3	Both	You should tune this timeout based on the type of access to the administration console (whether it is planned to use the Oracle WebLogic Server Administration Console from application tier clients or clients external to the application tier).
CRM Administration Console access	FW2	9001	HTTP / Administration Server and Enterprise Manager t3	Both	You should tune this timeout based on the type of access to the administration console (whether it is planned to use the Oracle WebLogic Server Administration Console from application tier clients or clients external to the application tier).
HCM Administration Console access	FW2	9401	HTTP / Administration Server and Enterprise Manager t3	Both	You should tune this timeout based on the type of access to the administration console (whether it is planned to use the Oracle WebLogic Server Administration Console from application tier clients or clients external to the application tier).
IC Administration Console access	FW2	9801	HTTP / Administration Server and Enterprise Manager t3	Both	You should tune this timeout based on the type of access to the administration console (whether it is planned to use the Oracle WebLogic Server Administration Console from application tier clients or clients external to the application tier).
BI Administration Console access	FW2	10201	HTTP / Administration Server and Enterprise Manager t3	Both	You should tune this timeout based on the type of access to the administration console (whether it is planned to use the Oracle WebLogic Server Administration Console from application tier clients or clients external to the application tier).
Node Manager	n/a	5556	TCP/IP	n/a	n/a

Table 2–2 (Cont.) Ports Used

Type	Firewall	Port and Port Range	Protocol / Application	Inbound / Outbound	Other Considerations and Timeout Guidelines
Access Server access	FW2	6021	OAP	Inbound	For actual values, see "Firewalls and Ports" in Oracle Fusion *Middleware Enterprise Deployment Guide for Oracle Identity Management*.
Identity Server access	FW2	6022	OAP	Inbound	
Database access for Oracle BI Server and Oracle BI Publisher JDBC Data Sources	FW2	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 Oracle BI
Database access	FW3	1521	SQL*Net	Both	Timeout depends on all database content and on the type of process model used for Oracle Fusion Applications.
Coherence for deployment	n/a	8088 and 8089 Range: 8000 - 8090		n/a	n/a
Oracle Internet Directory access	FW3	389	LDAP	Inbound	You should tune the directory server's parameters based on load balancer, and not the other way around.
Oracle Internet Directory access	FW3	636	LDAP SSL	Inbound	You should tune the directory server's parameters based on load balancer, and not the other way around.
JOC for OWSM	n/a	9991	TCP/IP	n/a	n/a

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

2.8 Clock Synchronization

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

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

3

Setting Up the Database Tier

This chapter provides information on how the database tier is implemented in an enterprise deployment topology.

This chapter includes the following topics:

- Section 3.1, "Understanding the Database in the Enterprise Deployment Topology"
- Section 3.2, "Setting Up the Database"
- Section 3.3, "Creating and Starting the Database Services"
- Section 3.4, "Loading the Oracle Fusion Applications Repository into the Oracle RAC Database"
- Section 3.5, "Backing Up the Database"

3.1 Understanding the Database in the Enterprise Deployment Topology

The Oracle Fusion Applications reference enterprise deployment topology uses a single database for the following components:

- Oracle Fusion Applications metadata
- Oracle Fusion Applications transactional data
- Oracle Transactional Business Intelligence data
- Technology stack data, such as Oracle SOA Suite, Oracle Enterprise Manager Fusion Middleware Control, Oracle WebCenter, and Oracle Essbase.
- Oracle Secure Enterprise Search data

Implementing Oracle Business Intelligence Data Warehouse requires a separate database for following components:

- Data Warehouse Administration Console (DAC)
- Informatica
- a Data Warehouse

For the enterprise topology, Oracle Real Application Clusters (Oracle RAC) databases are highly recommended. You must set up these databases before you can install and configure the Oracle Fusion Applications components. You install the Oracle Fusion Applications and Oracle Fusion Middleware metadata repositories into existing databases using the Fusion Applications Repository Creation Utility (Fusion Applications RCU).

3.2 Setting Up the Database

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

- Database Host Requirements
- Supported Database Versions
- Minimum Database Configuration Parameters

Note: When creating the database, ensure that the length of the Oracle System ID (SID) does not exceed eight (8) characters.

For example:

- *SID: abcd12345* is invalid
- *SID: abcd123* is valid

3.2.1 Database Host Requirements

Note the following requirements for the hosts *FUSIONDBHOST1* and *FUSIONDBHOST2* in the data tier:

- **Oracle Clusterware**

  For Oracle Database 11*g* Release 2 (11.2.0.2) for Linux, refer to the *Oracle Database Installation Guide*.

- **Oracle Real Application Clusters**

  For Oracle RAC 11*g* Release 2 (11.2.0.2) for Linux or Oracle Database 10*g* Release 2 (10.2) for Linux, refer to the *Oracle Database Installation Guide*.

- **Oracle Automatic Storage Management** (optional)

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

3.2.2 Supported Database Versions

Oracle Fusion Applications requires the presence of a supported database and schemas. To check if your database is certified or to see all certified databases, refer to the Oracle Fusion Applications system requirements and supported platforms documentation.

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

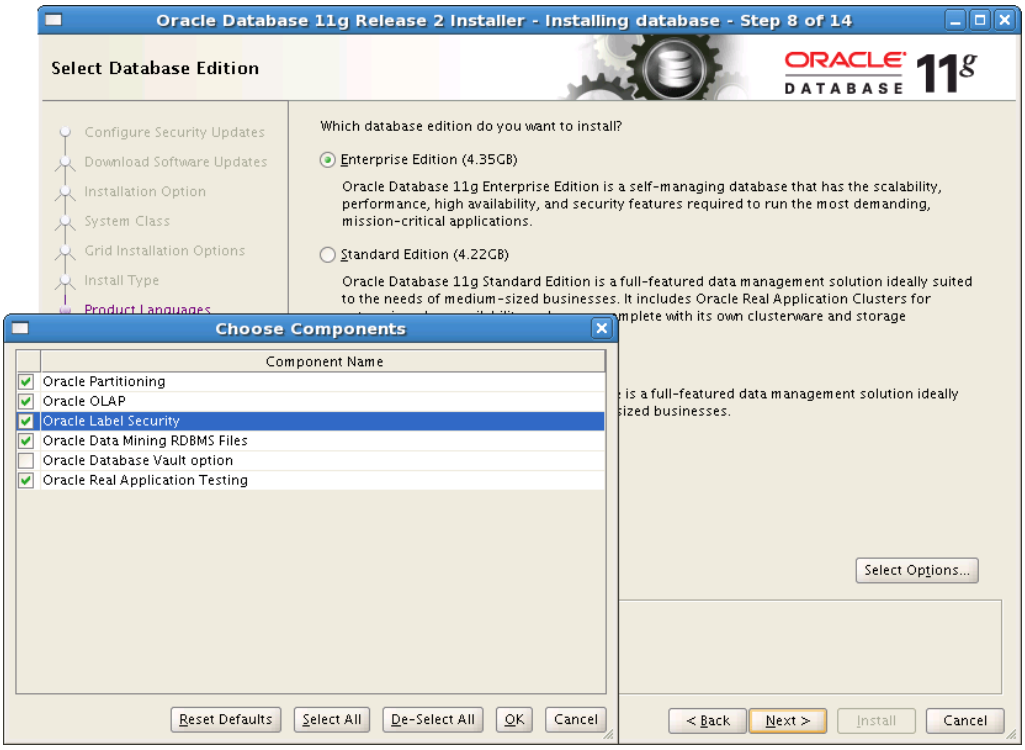
```
SQL> SELECT VERSION FROM SYS.PRODUCT_COMPONENT_VERSION WHERE PRODUCT LIKE
'Oracle%';
```

Notes:

- The database you use as the Oracle Fusion Applications supporting database must support the AL32UTF8 character set.
- When installing the database, please ensure that **Oracle Label Security** is enabled, as shown in Figure 3–1. In the case of an Oracle RAC installation, the Oracle Label Security should be enabled on all the nodes.

You enable label security in the Select Database Edition screen, shown in Figure 3–1, using Oracle Universal Installer.

Figure 3–1 Oracle Label Security



3.2.3 Minimum Database Configuration Parameters

Table 3–1 shows the recommended minimum init.ora parameters for an Oracle database. The database shipped with the Oracle Fusion Applications software contains this configuration. When you run the Fusion Applications RCU, its prerequisite check feature checks to see that the database meets these minimum requirements.

Table 3–1 Minimum Requirements for Database Configuration

INST_ID Parameter	Value
_b_tree_bitmap_plans	FALSE
audit_trail	NONE
compatible	11.2.0

Table 3–1 (Cont.) Minimum Requirements for Database Configuration

INST_ID Parameter	Value
db_files	1024
db_recovery_file_dest_size	2147483648
db_writer_processes	1
disk_asynch_io	FALSE
fast_start_mttr_target	3600
filesystemio_options	Setall
job_queue_processes	10
log_buffer	10485760
log_checkpoints_to_alert	TRUE
max_dump_file_size	10M
memory_target	unset or N/A
nls_sort	BINARY
open_cursors	500
pga_aggregate_target	>= 4294967296 (4GB)
plsql_code_type	NATIVE
processes	5000
session_cached_cursors	500
sga_target	>=9663676416 (9GB)
trace_enabled	FALSE
undo_management	AUTO

For example, to use the `SHOW PARAMETER` command using SQL*Plus to check the value of the initialization parameter:

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

```
SQL> SHOW PARAMETER processes
```

2. Set the initialization parameter using the following commands:

```
SQL> ALTER SYSTEM SET processes=5000 SCOPE=SPFILE;

SQL> ALTER SYSTEM SET open_cursors=500 SCOPE=SPFILE;
```

3. Restart the database.

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

3.3 Creating and Starting the Database Services

Oracle recommends using the Oracle Enterprise Manager Fusion Middleware Control Cluster Managed Services screen or SQL*Plus to create database services that client applications will use to connect to the database.

To configure this using SQL*Plus:

1. Use the `CREATE_SERVICE` subprogram to create the database service.

   Log in to SQL*Plus as the sysdba user and run the following command:

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

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

   ```
   prompt> srvctl add service -d fusiondb -s crm.mycompany.com -r fusiondb1
   fusiondb2
   ```

3. Start the service using `srvctl`:

   ```
   prompt> srvctl start service -d fusiondb -s crm.mycompany.com
   ```

4. Verify that the *crm* service is running on instance(s) *fusiondb1* and *fusiondb2*:

   ```
   prompt> srvctl status service -d fusiondb
   ```

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

Oracle recommends that a specific database service be used for a product suite even when they share the same database. It is also recommended that the database service used is different than the default database service. For example, for Oracle Fusion Customer Relationship Management the database would be *crmdb*.mycompany.com and the default service is one with the same name. The Oracle Fusion Customer Relationship Management install is configured to use the service *crm*.mycompany.com.

3.3.1 Updating the Kernel Parameters

This section describes how to update the kernel parameters for Linux before the database is installed.

To update the parameters:

1. Log in as root and add or edit the following values in the `/etc/sysctl.conf` file:

   ```
   fs.file-max = 6815744
   kernel.shmall = 2097152
   kernel.shmmax = 2147483648
   kernel.shmmni = 4096
   kernel.sem = 250 32000 100 128
   net.core.rmem_default = 4194304
   net.core.rmem_max = 4194304
   net.core.wmem_default = 262144
   net.core.wmem_max = 1048576
   net.ipv4.ip_forward = 0
   ```

```
net.ipv4.conf.default.rp_filter = 1
tcp.ipv4.tcp_wmem = 262144 262144 262144
tcp.ipv4.tcp_rmem = 4194304 4194304 4194304
fs.aio-max-nr = 1048576
net.ipv4.ip_local_port_range = 9000 65000
```

2. Execute the following command to activate the changes:

```
/sbin/sysctl -p
```

3.4 Loading the Oracle Fusion Applications Repository into the Oracle RAC Database

Before loading the Oracle Fusion Applications repository into a database, you must apply database patch 10220058 in order to run the Oracle Fusion Applications Repository Creation Utility (Fusion Applications RCU) with Oracle Database 11*g* Enterprise Edition Release 11.2.0.2.0. To find the patch, go to My Oracle Support (https://support.oracle.com) and click the **Patches & Updates** tab.

The Fusion Applications RCU components are included in the zipped Fusion Applications RCU file delivered in the provisioning framework. Unzip the file to the *RCU_HOME* location on the *FUSIONDBHOST1* machine. For example, *ORACLE_BASE*/rcu.

Once you have the Fusion Applications RCU installed, do the following:

1. Copy all the required dump files locally on *FUSIONDBHOST1* (for example, to /tmp):

```
FUSIONDBHOST1> cd RCU_HOME/rcu/integration/fusionapps
FUSIONDBHOST1> cp export_fusionapps_dbinstall.zip /tmp
FUSIONDBHOST1> cd RCU_HOME/rcu/integration/biapps/schema
FUSIONDBHOST1> cp otbi.dmp /tmp
FUSIONDBHOST1> cd /tmp
FUSIONDBHOST1> unzip export_fusionapps_dbinstall.zip
```

> **Note:** When running Fusion Applications RCU for Oracle RAC, you must copy the export_fusionapps_dbinstall.zip file as well as otbi.dmp to all the nodes of the Oracle RAC.

2. Create the incident_logs directory on *FUSIONDBHOST1*:

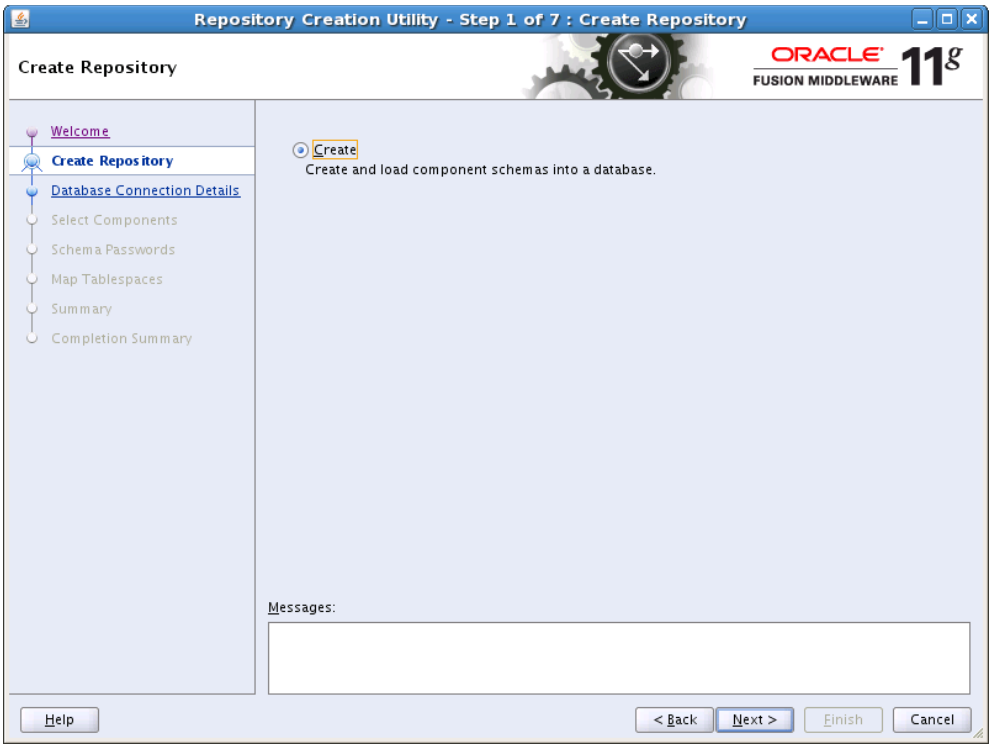
```
FUSIONDBHOST1> cd ORACLE_HOME
FUSIONDBHOST1> mkdir incident_logs
```

3. Start Fusion Applications RCU from the /bin directory in the Fusion Applications RCU home directory:

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

4. In the Welcome screen (if displayed), click **Next**.
5. In the Create Repository screen, shown in Figure 3–2, select **Create** to load component schemas into a database. Click **Next**.

Figure 3–2 Create Repository Screen

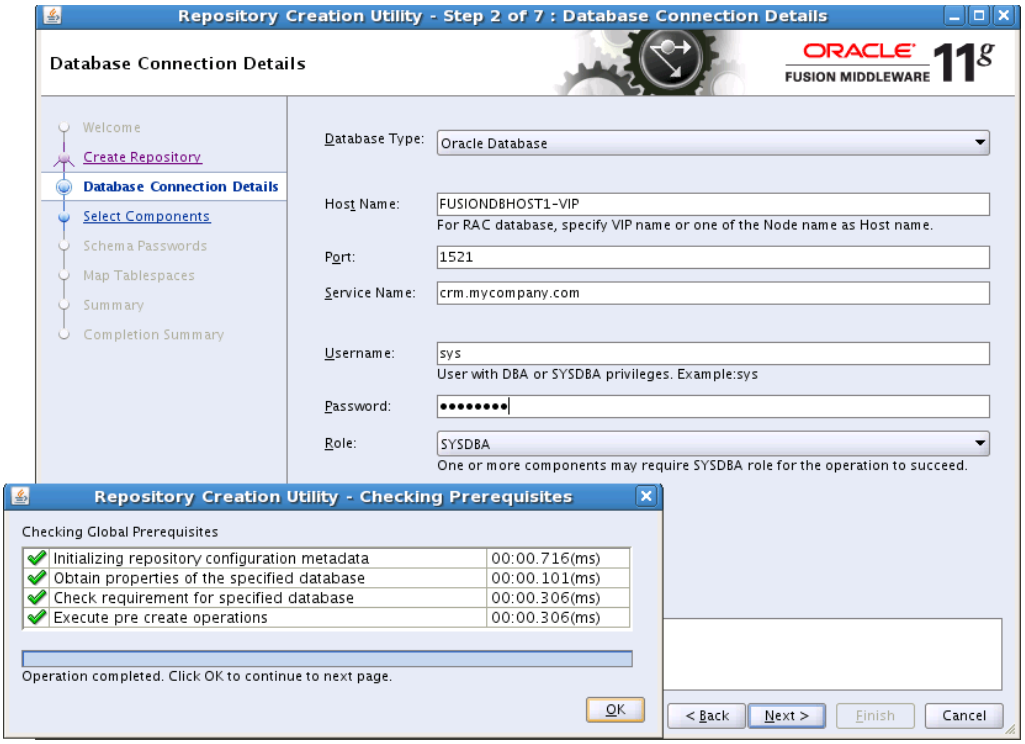


6. In the Database Connection Details screen, shown in Figure 3–3, enter connect information for your database:

   - **Database Type**: Select **Oracle Database**
   - **Host Name**: Specify the name of the node on which the database resides. For the Oracle RAC database, specify the VIP name or one of the node names as the host name: *FUSIONDBHOST1*-VIP
   - **Port**: Specify the listen port number for the database; for example 1521
   - **Service Name**: Specify the service name of the database (*crm*.mycompany.com)
   - **Username**: Specify the name of the user with DBA or SYSDBA privileges: SYS
   - **Password**: Enter the password for the SYS user
   - **Role**: Select the database user's role from the list: SYSDBA (required by the SYS user)

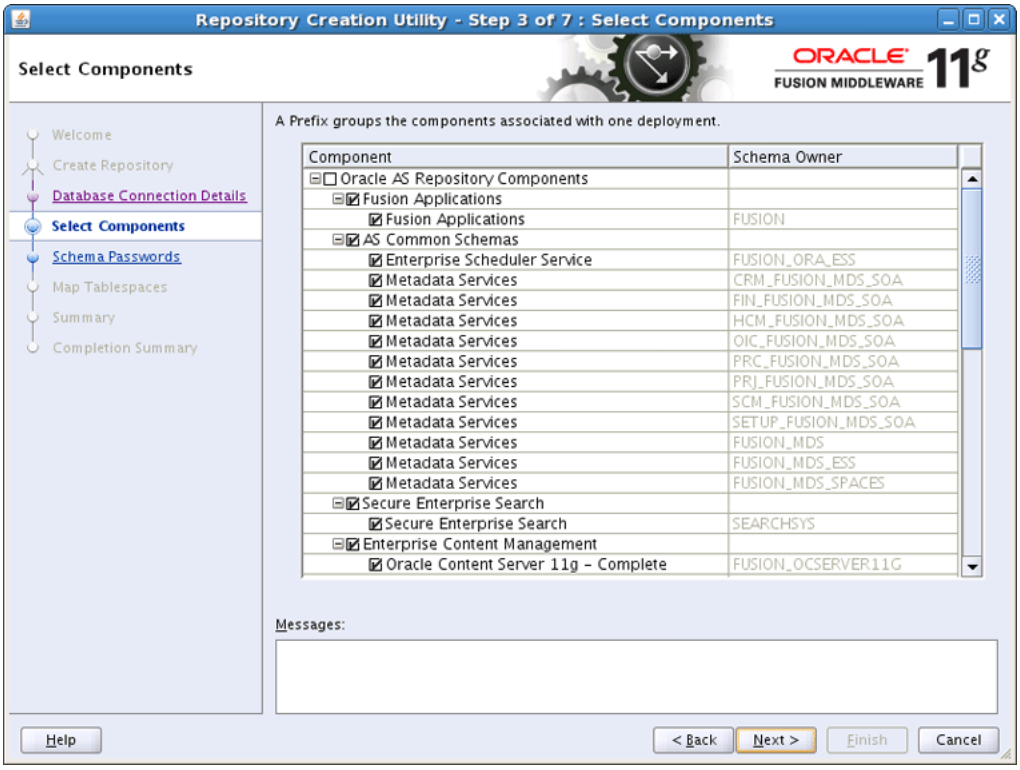
   Click **Next**.

Figure 3–3 Database Connection Details Screen



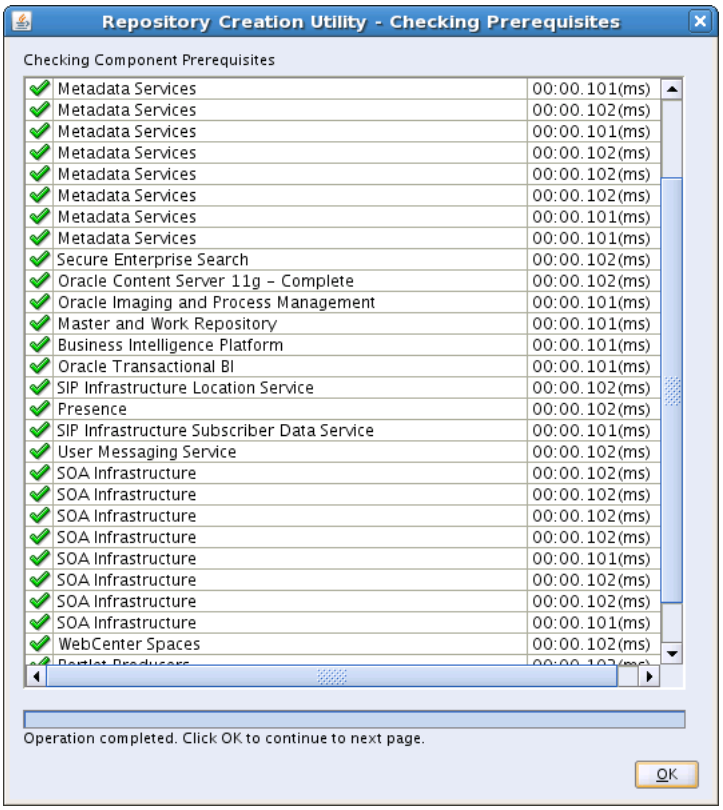
7. The Repository Creation Utility selects the required components automatically, as shown in Figure 3–4.

Figure 3–4 Select Components Screen



Click **Next**. The Repository Creation Utility checks the prerequisites, as shown in Figure 3–5.

Figure 3–5 Prerequisite Check



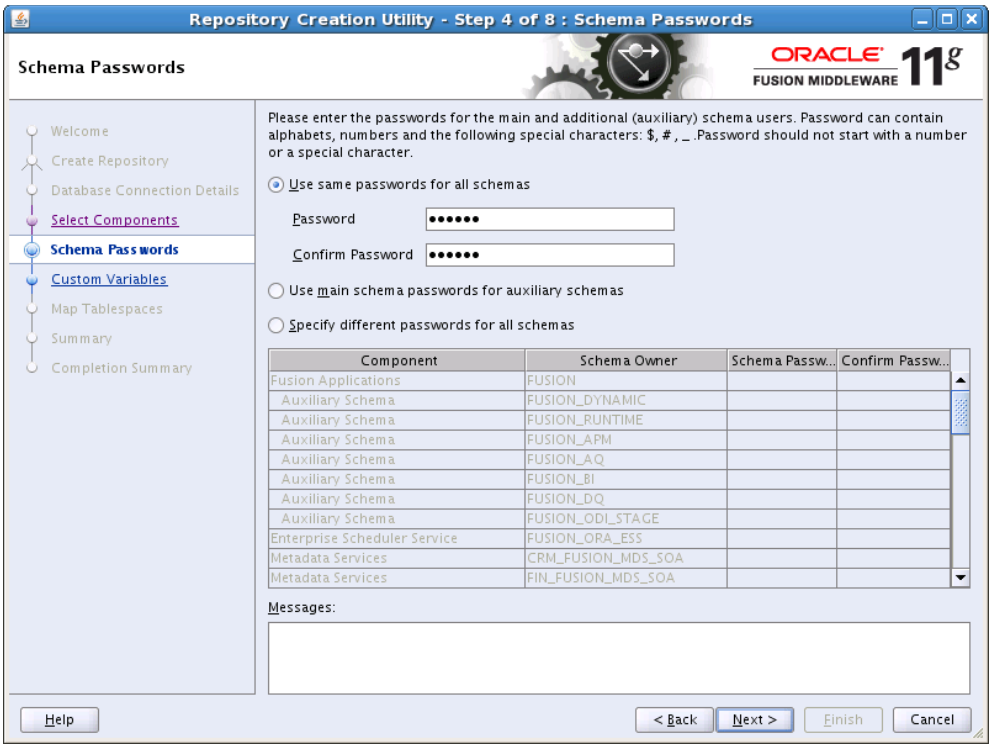
Click **OK**.

8. In the Schema Passwords screen, shown in Figure 3–6, enter passwords for the main and additional (auxiliary) schema users, and click **Next**.

   Note: For increased security, do the following:

   - Specify different schema passwords for all schemas
   - Ensure that all paswords are more than eight (8) characters in length.

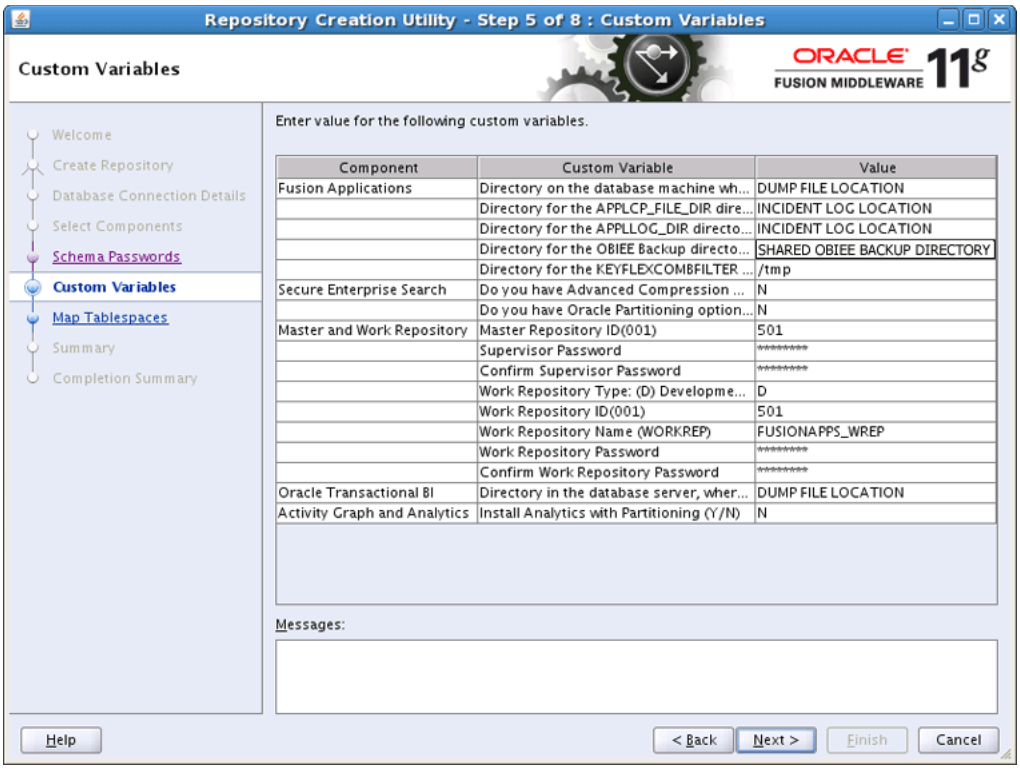
Figure 3–6 Schema Passwords Screen



9. In the Custom Variables screen, shown in Figure 3–7, enter the required values:
   - **DUMP FILE LOCATION** - `/tmp`
   - **INCIDENT LOG LOCATION** - *`ORACLE_HOME`*`/incident_logs`

Note: When specifying an Oracle Business Intelligence Enterprise Edition (OBIEE) backup directory, set it to be a shared directory with read/write permissions for both DAC database hosts.

Figure 3–7 Custom Variables Screen



10. In the Map Tablespaces screen, choose the tablespaces for the selected components, and click **Next**.
11. In the Summary screen, click **Create**.
12. In the Completion Summary screen, click **Close**.

Note: If you encounter any issues while using the Repository Creation Utility, check the logs at *RCU_HOME*`/rcu/log`.

Note: Oracle recommends using the database used for identity management to store the Oracle WSM policies. It is therefore expected to use the IM database information for the OWSM MDS schemas, which will be different from the one used for the rest of SOA schemas. To create the required schemas in the database, repeat the steps above using the IM database information, but select only "AS Common Schemas: Metadata Services" in the Select Components screen (Step 7).

3.5 Backing Up the Database

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

Backing up the database is for the explicit purpose of quick recovery from any issue that may occur in the further steps. You can choose to use your backup strategy for the database for this purpose or simply make a backup using operating system tools or RMAN for this purpose. It is recommended that you use Oracle Recovery Manager for

the database, particularly if the database was created using Oracle ASM. If possible, a cold backup using operating system tools such as tar can also be performed.

4

Using the Provisioning Process to Install Components for an Enterprise Deployment

This chapter describes the provisioning process used to install and configure components specifically required for an enterprise deployment.

For general information about provisioning and installation, see the "Overview" chapter in the *Oracle Fusion Applications Installation Guide*.

This chapter includes the following topics:

- Section 4.1, "Understanding Provisioning"
- Section 4.2, "Prerequisites for Using the Provisioning Process"
- Section 4.3, "Installing Components"
- Section 4.4, "Configuring Components"
- Section 4.5, "Performing Post-Provisioning Validation"

4.1 Understanding Provisioning

Provisioning is the entire set of operations required to install, configure, and deploy applications product offerings from a system point of view. It performs these operations:

- **Install** - operations related to laying down all the component needed to create an Oracle Fusion Applications environment.
- **Configure** - the tailoring of components based on the applications topology, the creating of managed server instances and cluster members, and the updating of endpoints and virtual hosts.
- **Deploy** - process that starts the managed servers and clusters and facilitates the actual use of product offerings.

> **Note:** Provisioning does not supply users, tenants, or hardware.

For more information about Oracle Fusion Applications architecture, see "Key Oracle Fusion Applications Concepts" in the *Oracle Fusion Applications Administrator's Guide*.

4.2 Prerequisites for Using the Provisioning Process

Before starting the provisioning process, you must do the following:

- Make sure you first install the Oracle Identity Management stack for Oracle Fusion Applications. Follow the instructions in *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Identity Management* to install and configure these components.

  Keep a record of the configuration details. You must supply them to the Provisioning Wizard when you create your provisioning plan. For more information, see Section 4.3.2.15, "Identity Management Configuration Screen."
- Make sure you obtain the certificates file from the Oracle Identity Management installation. The installation contains all the Oracle Identity Management certs.
- Make sure all the virtual IPs shown in Table 2–1 in Chapter 2, "Network Configuration" have been created before you start the provisioning process. (The VIPs are required for the scale-out chapters that follow, and not for provisioning.)
- Make sure the SMTP server is preinstalled, and that its host name and port value are available.
- (Optional) If the user-created user needs to be a superuser, create a user for the superuser (perform these steps on the OIDHOST of the Oracle Identity Management setup):

  1. Create a `SuperUser.ldif` file and include the following data:

     ```
     dn: cn=FAAdmin,cn=Users,dc=MyCompany,dc=com
     objectclass: top
     objectclass: person
     objectclass: organizationalPerson
     objectclass: inetOrgPerson
     objectclass: orclUser
     objectclass: orclUserV2
     cn: FAAdmin
     sn: FAAdmin
     description: This user will be provisioned for super admin group
     givenname: FAAdmin
     orclsamaccountname: FAAdmin
     uid: FAAdmin
     userpassword: welcome1
     ```

  2. Run the following command to load the file into the LDAP:

     ```
     ldapadd -h hostname -p port -D cn=username -w password -c -x -f
     SuperUser.ldif
     ```

     This will create a superuser, *FAAdmin*, with a *welcome1* password. These values will be used later in the chapter.

  3. Do the following to make the superuser SSO-enabled:

     a. Create an `inputfile.config` file with following values:

        ```
        IDSTORE_HOST: IDSTORE_HOST
        IDSTORE_PORT: IDSTORE_PORT
        IDSTORE_ADMIN_USER: IDSTORE_ADMIN_USER
        IDSTORE_USERSEARCHBASE: IDSTORE_USERSEARCHBASE
        IDSTORE_GROUPSEARCHBASE: IDSTORE_GROUPSEARCHBASE
        PASSWORD_EXPIRY_PERIOD: 7300
        ```

     Note: The value "7300" is an example `PASSWORD_EXPIRY_PERIOD`.

b. Run the following command (note that *ORACLE_HOME*, *MW_HOME*, and *JAVA_HOME* need to be set):

```
./idmConfigTool.sh -upgradeLDAPUsersForSSO input_file=inputfile.config
log_level=ALL log_file=upgradeLDAPUsersForSSO.out dump_params=true
```

Note: The `idmConfigTool.sh` script can be found in the Oracle Identity Management installation.

4.3 Installing Components

This section covers the following topics:

- Section 4.3.1, "Creating the Installation Environment"
- Section 4.3.2, "Creating a New Provisioning Plan"
- Section 4.3.3, "Running the Provisioning Commands to Install Components"

4.3.1 Creating the Installation Environment

Oracle Fusion provisioning repositories consist of multiple installers from Oracle Fusion Middleware and Oracle Fusion Applications. In order to run the Fusion provisioning process, these installers must be in a predefined directory structure.

This section includes the following topics:

- Downloading the Provisioning Repository
- Installing the Provisioning Framework Bits

4.3.1.1 Downloading the Provisioning Repository

A zipped provisioning repository is included in the Oracle Fusion Applications Product Media Pack. See "Obtaining the Software" in *Oracle Fusion Applications Installation Guide* for instructions on how to get it.

Extract the contents of all the zipped files to the same target directory that is on a shared/network drive. By default, the installers are located in *repository_location*`/installers`.

4.3.1.2 Installing the Provisioning Framework Bits

The provisioning framework supplies the components needed to orchestrate the provisioning process. Once set up, this framework retrieves the components and installers silently when they are required.

4.3.1.2.1 Running the Provisioning Installer Run the installer from the directory where you created the provisioning repository. For example: *repository_location*`/repository/installers/faprov/Disk1`.

Note: If you are running a fresh install or are re-running the installer after cleaning up previously installed products, ensure that no `/etc/oraInst.loc` file exists.

To run the installer:

```
CRMHOST1> ./runInstaller
```

When prompted, enter the following JRE/JDK location:

```
repository_location/repository/jdk6
```

Use the screen information in Table 4–1 as a guide when running the installer.

> **Note:** In the case of a clean host, that is, one where the `/etc/oraInst.loc` file does not exist, the oraInventory creation panels will display prior to the start of the Provisioning Wizard. In addition, a confirmation dialog asking you to execute `oracleRoot.sh` will display at the end of the installation.

Table 4–1 Provisioning Installer Screens

Screen Name	Description
Welcome	The standard Welcome screen is read-only and appears each time you start the provisioning framework installer. No action is required. Click **Next** to continue.
Prerequisite Checks	Analyzes the host computer to ensure that specific operating system prerequisites have been met. If any prerequisite check fails, the screen displays a short error message at the bottom. Fix the error and click **Retry**. If you want to ignore the error or warning message, click **Continue**. Click **Abort** to stop the prerequisite check process for all components. Click **Next** to continue.
Specify Installation Location	Specify a location where you want to install the provisioning framework (*ORACLE_BASE*`/repository`). This is the location where the Provisioning Wizard and the start commands for provisioning (`runProvisioning`) are installed. The Oracle Fusion Applications Provisioning framework must be installed on a shared disk in a location that is accessible to all hosts to be provisioned. Click **Next** to continue.
Installation Summary	Summarizes the selections you have made during this installation session. To change this configuration before installing, select one of the screens from the left navigation pane. Click Save to create a text file (response file) to use if you choose to perform the same installation at a later date. Click **Install** to continue installing this configuration.
Installation Progress	The progress indicator shows the percentage of the installation that is complete and indicates the location of the installation log file. Click **Next** when the progress indicator shows 100 percent.
Installation Complete	Summarizes the installation just completed. If you want to save the details to a text file, click Save and indicate a directory where you want to save the file. Click Finish to dismiss the screen and exit the installer.

/provisioning Directory Structure:

After installing the provisioning framework, the directories in *ORACLE_BASE*/repository/provisioning should be the following:

```
ant  bin  labelInfo.txt  lib  provisioning-build  provisioning-plan
template  util
```

Note: Installation logs are located in the /oraInventory directory.

Move or copy the directories and files from the *repository_location* to *ORACLE_BASE*/repository:

```
CRMHOST1> mv repository_location/repository/* ORACLE_BASE/repository
```

4.3.2 Creating a New Provisioning Plan

Before provisioning the Oracle Fusion Applications enterprise deployment environment, you must generate the provisioning plan, which will serve as the input for the actual provisioning process. You generate the provisioning plan by completing a number of wizard interview screens to collect the configuration details for your provisioning plan and save the plan in a location that is accessible to the provisioning installers. Be sure to make a note of the provisioning plan file name and location, as you must supply these when you run the physical installation.

Before launching the provisioning wizard, set JAVA_HOME and PATH. For example:

```
CRMHOST1> export JAVA_HOME=ORACLE_BASE/repository/jdk6

CRMHOST1> export PATH=$JAVA_HOME/bin:$PATH
```

Launch the provisioning wizard from any host in the enterprise deployment environment:

```
CRMHOST1> cd ORACLE_BASE/repository/provisioning/bin

CRMHOST1> ./provisioningWizard.sh
```

The Oracle Fusion Applications Provisioning Wizard is launched and the Welcome screen displays. The screen is read-only and displays each time you start the Wizard.

Click **Next**.

4.3.2.1 Installation Options Screen

In this screen, shown in Figure 4–1, select only the following task from the list of options:

Create a New Applications Environment Provisioning Plan - create a provisioning plan for a new Oracle Fusion Applications environment.

Figure 4–1 Installation Options Screen

Click **Next** to continue.

4.3.2.2 Specify Security Updates Screen

In this screen, you can set up a notification preference for security-related updates and installation-related information from Oracle Support.

- **Email** - specify your email address to have updates sent by this method.
- **I wish to receive security updates via My Oracle Support** - specify your **My Oracle Support Password** to have updates posted to your account.

Click **Next** to continue.

4.3.2.3 Provisioning Configurations Screen

This screen, shown in Figure 4–2, enables you to select the Oracle Fusion Customer Relationship Management options to configure.

Select only the options shown in Figure 4–2:

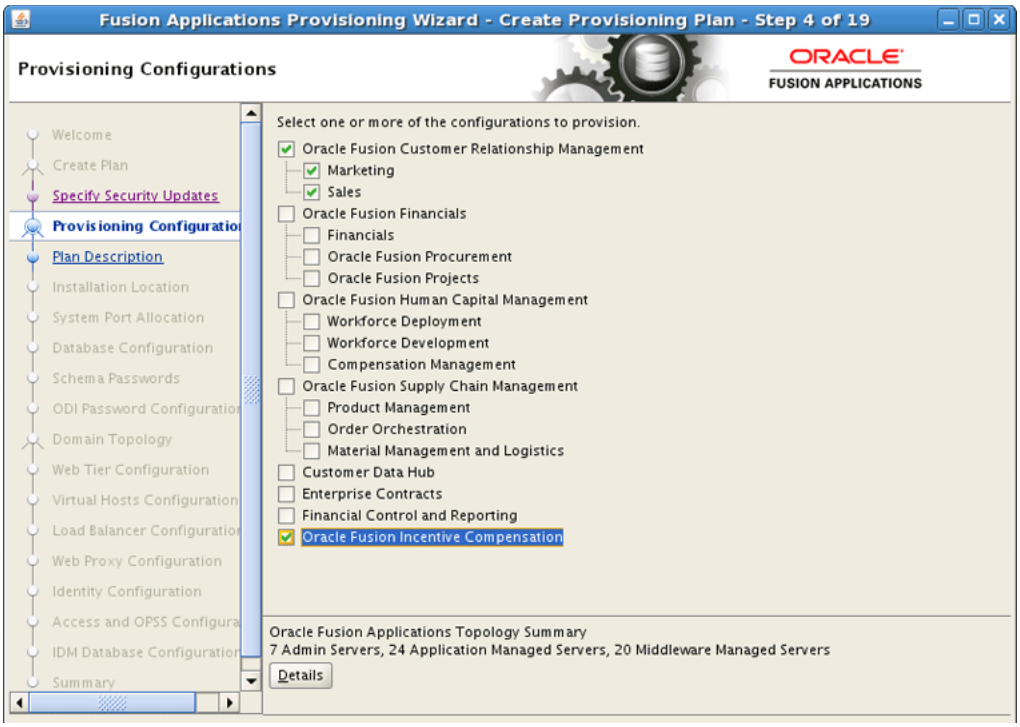
- Oracle Fusion Customer Relationship Management

  When selected, the Marketing and Sales options are automatically selected.

- Oracle Fusion Incentive Compensation

The information in the message pane displays a cumulative estimate of the number of managed servers made available based on the offerings you selected. Click **Details** to see a breakdown of servers by domain.

Figure 4–2 Provisioning Configurations Screen



Click **Next** to continue.

4.3.2.4 Plan Description Screen

This *optional* screen lets you enter descriptive information to identify this plan, or create another version. This information becomes part of the plan summary document, and is listed under the Global settings on the **Summary** screen. It does not affect the content of your plan.

Update plan name and click **Next** to continue.

4.3.2.5 Installation Location Screen

In this screen, shown in Figure 4–3, specify credentials for the node manager and supply the location of the various directories required for installation and configuration actions.

Use the values shown in the screen for your installation.

- **Node Manager Credentials** options - Add the values for the Node Manager credentials, which are used by Node Manager to start the managed server.
- **Installers Directory Location** - Specify the location of the repository you created. For example, *ORACLE_BASE*`/repository`.
- **Oracle Fusion Application Home** - The `root` directory of all Oracle Fusion Applications and Oracle Fusion Middleware products. Typically, this location is on a shared disk, *ORACLE_BASE*`/products`.
- **Application Configuration Directory** - Specify the path of the root directory where you want to write and manage the configuration files for all the domains, and from where the Administration Servers are started. Typically, this location is on a shared disk, *ORACLE_BASE*`/config`. (Note that *ORACLE_BASE*`/config` should be empty.)

- **Enable Local Domain Configuration** - Enable this option. When enabled, all the Managed Servers will run locally; only the Administration Server will run from the shared disk. Provisioning will run `pack` and `unpack`, and will create local domain directories.
- **Local Domain Config Directory** - Specify a local-drive location, for example, `/u02/local/oracle/config`. This field is required if you selected **Enable Local Application Configuration**.
- **WebGate Library Location** - Specify the location of the WebGate library: *ORACLE_BASE*`/repository/installers/webgate/lib`. Make sure that the directory you specifiy for the WebGate library location exists, and that the *libstdc++.so.5* and *libgcc_s.so.1* Open Source libraries are in that directory.

  Run the following commands to see if the *libstdc++.so.5* and *libgcc_s.so.1* libraries are already installed:

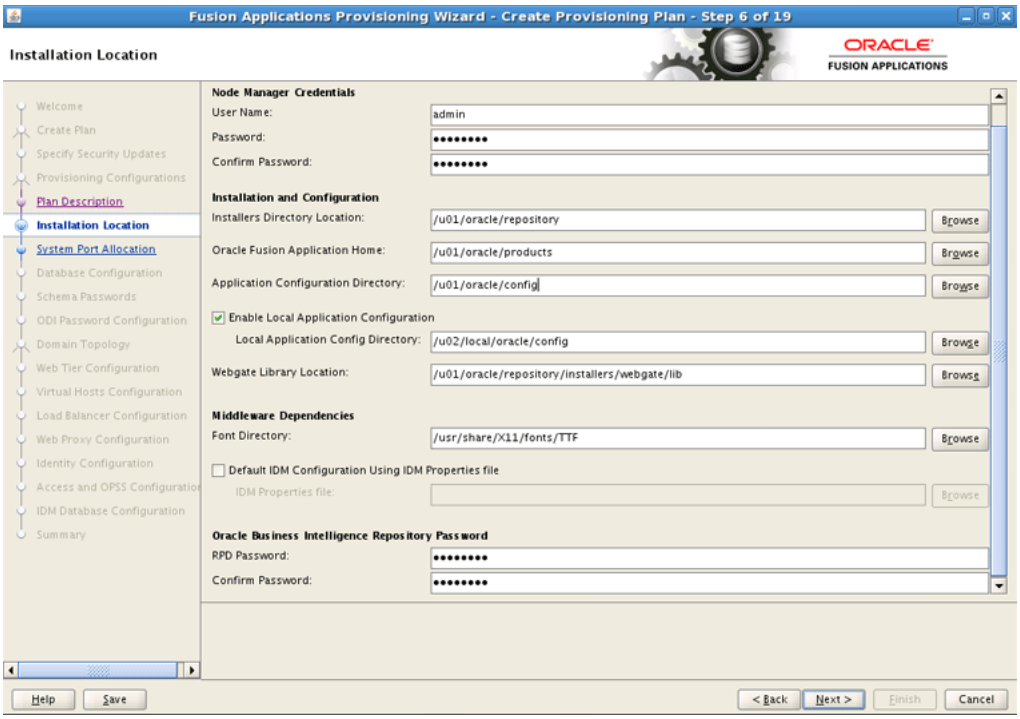
  ```
  CRMHOST1> /usr/bin/gcc --version

  CRMHOST1> /usr/bin/gcc --print-file-name=libstdc++.so.5

  CRMHOST1> /usr/bin/gcc --print-file-name=libgcc_s.so.1
  ```

  If the libraries are installed, copy them to *ORACLE_BASE*`/repository/installers/webgate/lib`.

  If they are not installed, download the appropriate GCC (C++ compiler) library sources from `http://gcc.gnu.org/` and compile them to obtain the libraries. For some operating systems, the required libraries may be available as installable packages from the support web sites of operating-system vendors.
- **Font Directory** - Enter the directory where the TrueType fonts are installed. The location varies on different operating systems, but is typically found at `/usr/share/X11/fonts/TTF`.
- **Default IDM Configuration Using IDM Properties file** - Check this box if you want the values on the **Identity Management Configuration** and the **Access and Policy Management Configuration** screens to default to the values in the Oracle Identity Manager properties file. Provide the full path to the file.
- **Oracle Business Intelligence Repository Password** options - Specify and confirm a password to allow access to the metadata repository (RPD) for both Oracle Business Intelligence Applications and Oracle Transactional Business Intelligence.

Note: Ensure that the WebGate library location already exists prior to entering its location in the Installation Location Screen.

Figure 4–3 Installation Location Screen



Click **Next** to continue.

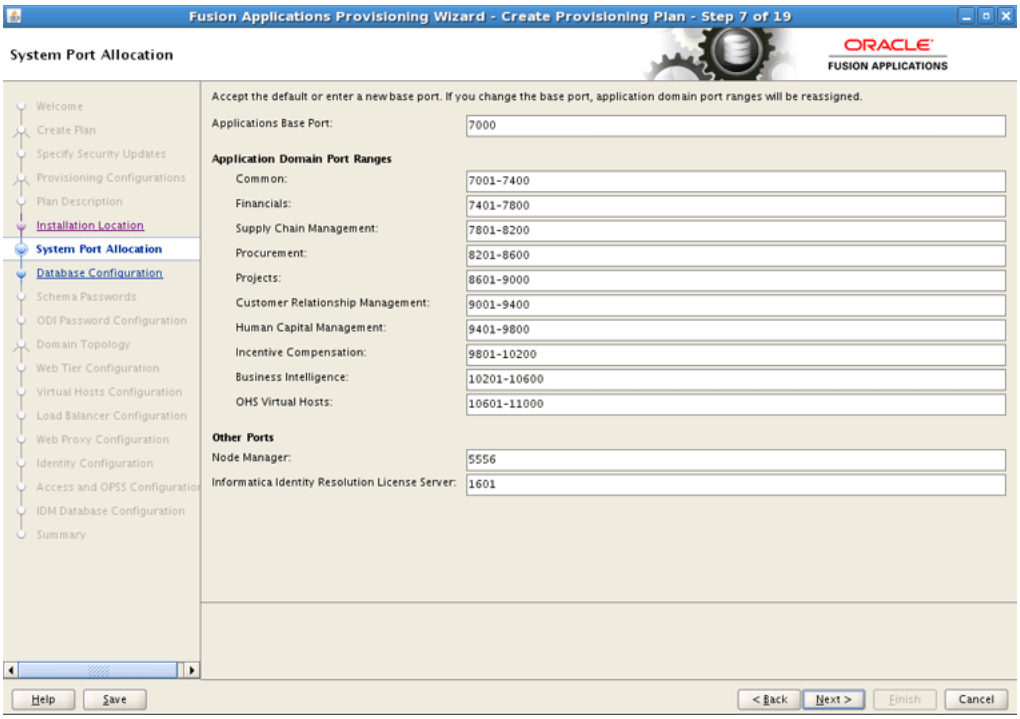
4.3.2.6 System Port Allocation Screen

In this screen, shown in Figure 4–4, accept the **Applications Base Port** value or enter a custom value. If you change the base port default, you must reset the domain port ranges accordingly. Port ranges must not overlap and must be stated as ascending values.

High and low port ranges are assigned by default to each product family per domain in the **Application Domain Port Ranges** list. The default range allotment for each product family is 399, with each family's range arranged in ascending order.

The **Other Ports** section contains the default value for the Node Manager Port.

Figure 4–4 System Port Allocation Screen



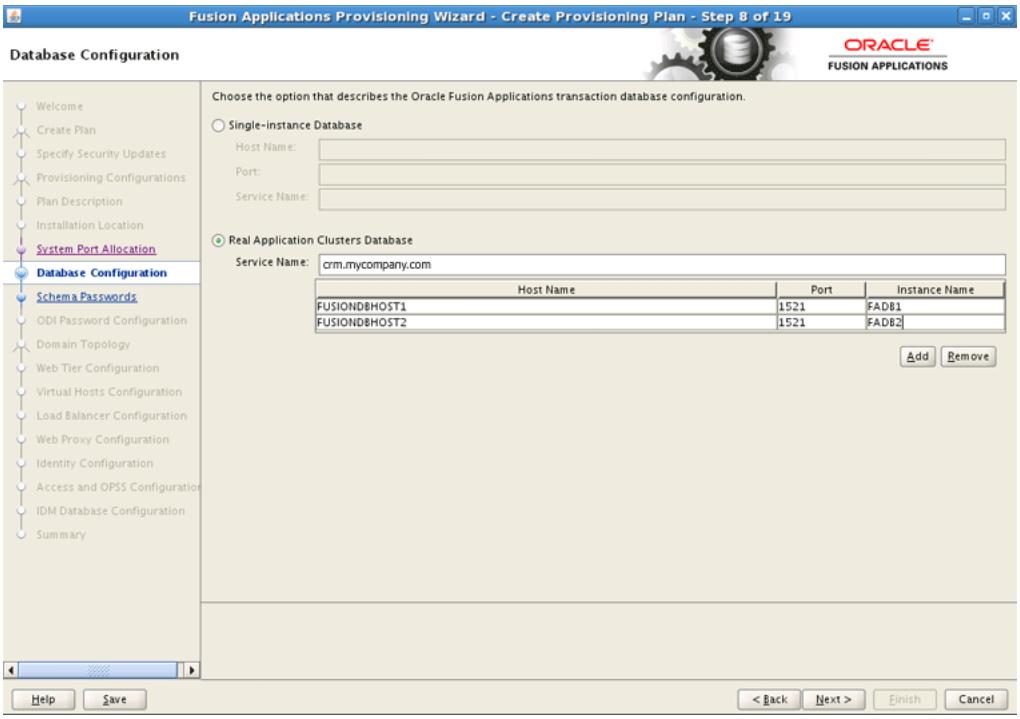
Click **Next** to continue.

4.3.2.7 Database Configuration Screen

In this screen, shown in Figure 4–5, click **Add** to create a line in the table for each instance in this database. Select a row and click **Remove** if you need to revise the table. Specify the following information for each instance:

- **Host Name** - the name of the Oracle RAC host for each instance.
- **Port** - listening port of the database.
- **Instance Name** - the Oracle RAC database instance name

Figure 4–5 Database Configuration Screen



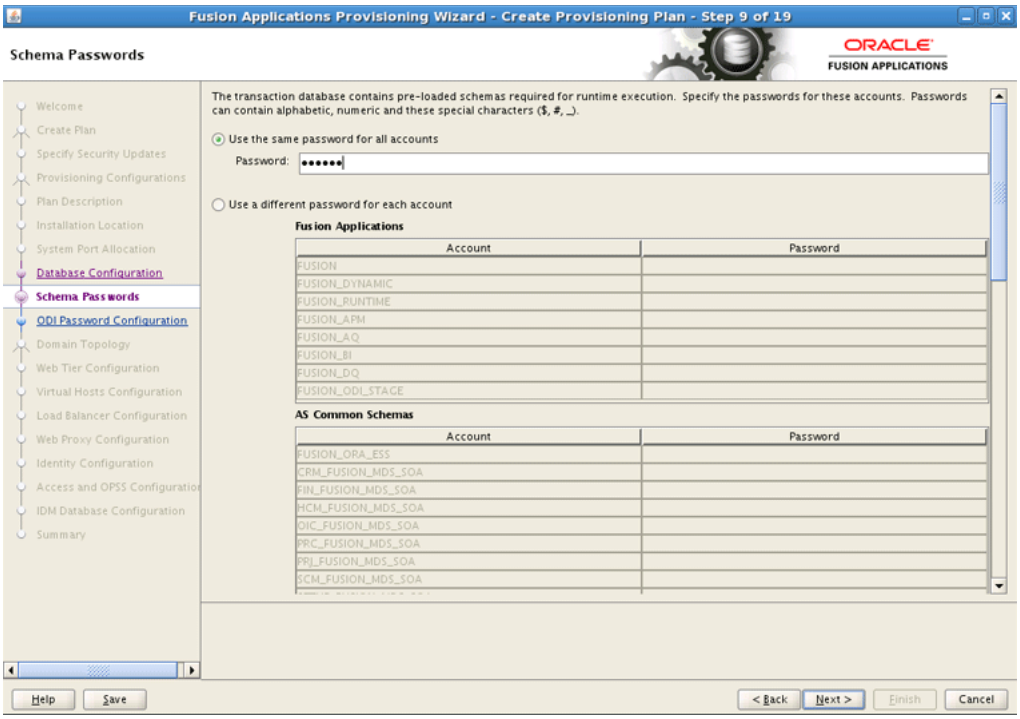
Click **Next** to continue.

4.3.2.8 Schema Passwords Screen

In this screen, shown in Figure 4–6, enter the same password for all the accounts or, if there are different passwords for each account, select **Use a different password for each account** and enter the passwords.

> **Note:** It is recommended to use a separate password for each account in the production deployment.

Figure 4–6 Schema Passwords Screen



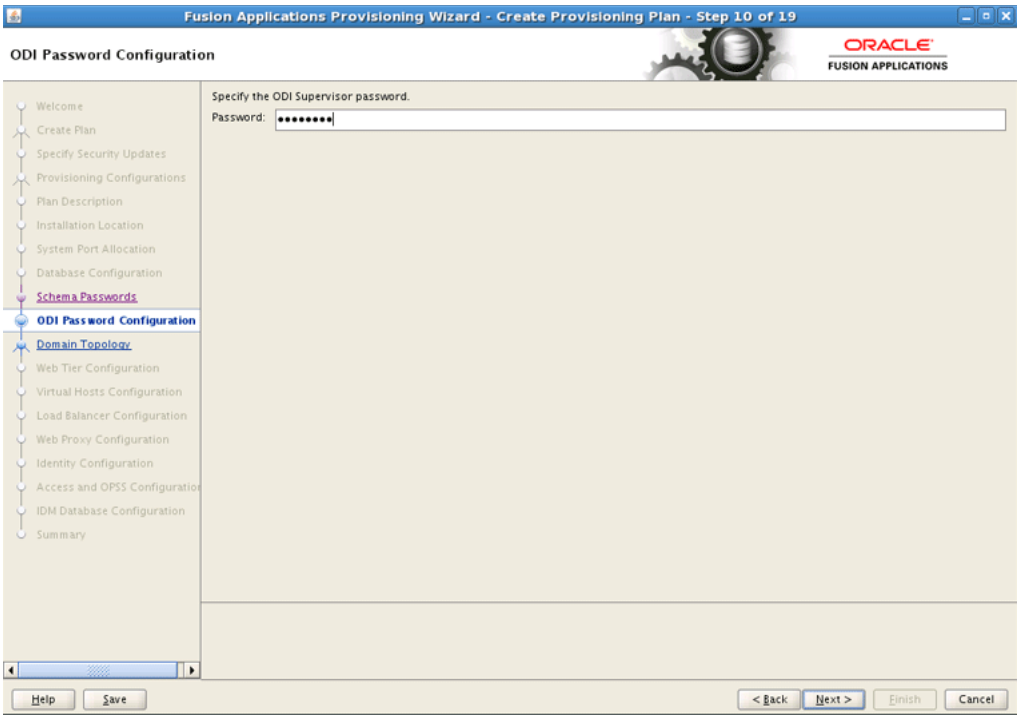
Click **Next** to continue.

4.3.2.9 ODI Password Configuration Screen

In this screen, shown in Figure 4–7, enter the Oracle Data Integrator Supervisor Password that was used when the Oracle Fusion Middleware Metadata Repository was loaded into the Oracle RAC database (see Figure 3–7 in Section 3.4, "Loading the Oracle Fusion Applications Repository into the Oracle RAC Database").

Click **Next** to continue.

Figure 4–7 ODI Password Configuration

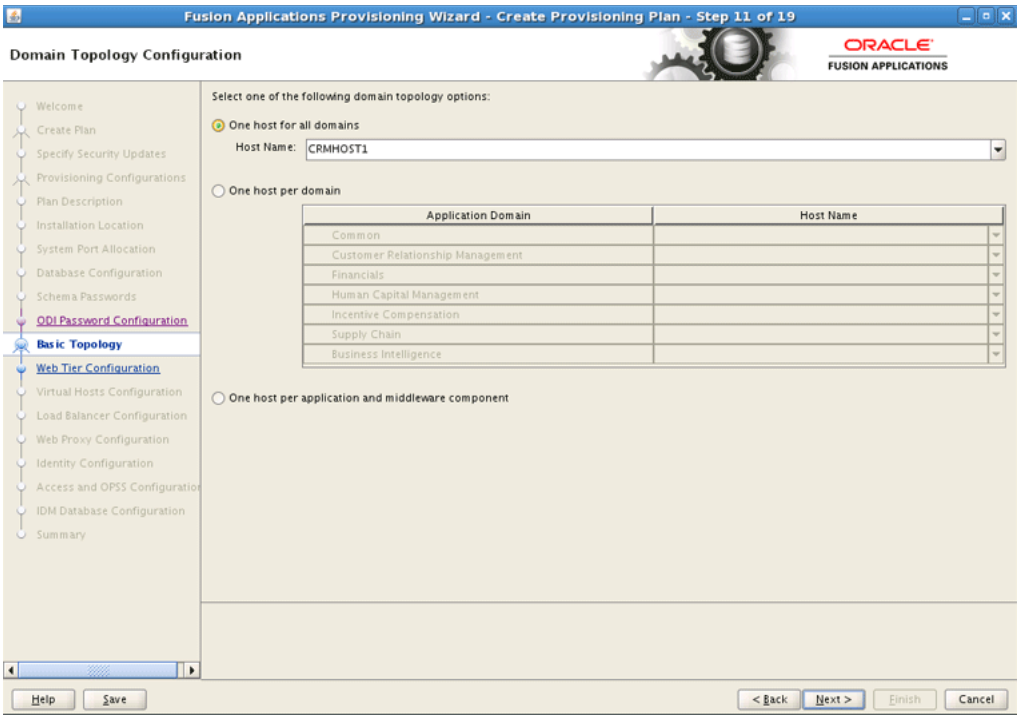


4.3.2.10 Domain Topology Configuration Screen

In this screen, shown in Figure 4–8, determine the flow for the remaining wizard interview screens.

- **One host for all domains** - select this option to specify a **Host Name** if there is only one host and the ports are not changing.
- **One host per domain** - select this option if the domains are to be split among several machines. Use the dropdown list to select a **Host Name** for each application domain to be created.
- **One host per application and middleware component** - select this option when there are different hosts and ports to be moodified.

Figure 4–8 Domain Topology Configuration Screen



4.3.2.11 Web Tier Configuration Screen

This screen, shown in Figure 4–9, allows you to create virtual hosts on a single Oracle Web Tier that are either port-based or name-based for each product family domain that is created during installation. Specify an internal and an external port. The values assigned during installation are derived from the default HTTP port you name on this screen.

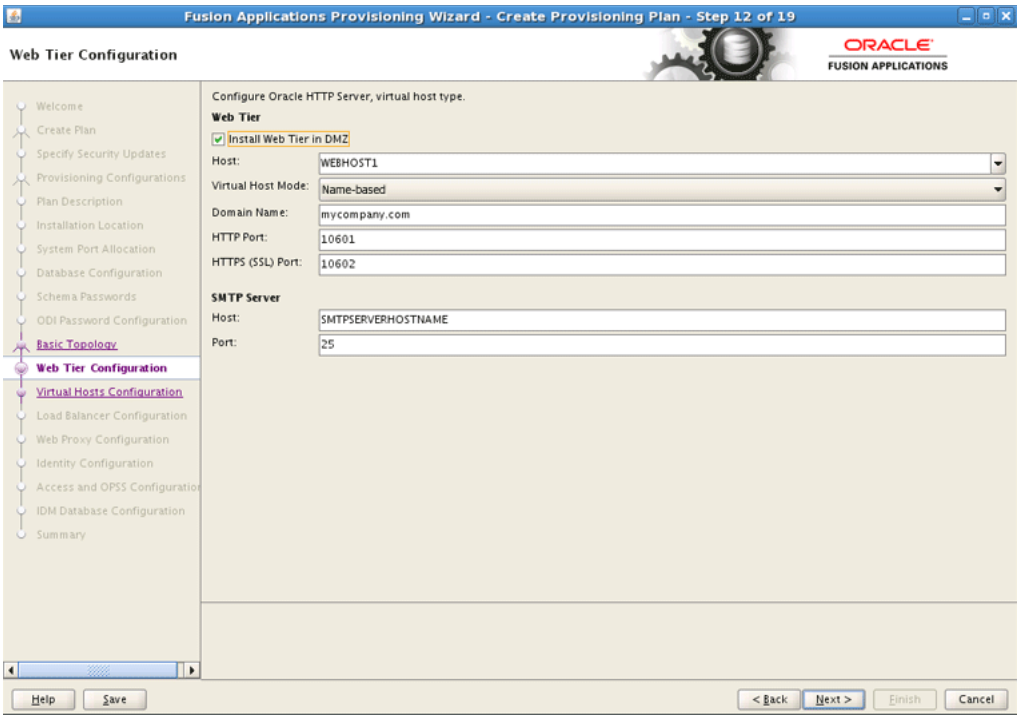
Web Tier

- **Install Web Tier in DMZ** - select this option if you set up a separate host for web tier installation. This host is set up as a demilitarized zone (DMZ), which does not have access to the shared file system. It cannot be used for any other host deployed, regardless of domain.
- **Host** - enter the name of the host where the Oracle HTTP server will be installed and configured.
- **Virtual Host Mode** - select **Name-based** to create new DNS entries to use as virtual hosts. For example, `fin.mycompany.com` and `crm.mycompany.com`.
- **Domain Name** - specify a domain name (only if you select a name-based virtual host). For example, `mycompany.com`.
- **HTTP Port** - default port for the Web Tier. Should not require operating system administrator privileges. Use the default values.
- **HTTP (SSL) Port** - secure port for the Web Tier. Should not require operating system administrator privileges. Use the default values.

SMTP Server

- **Host** - the pre-installed SMTP server host name.
- **Port** - the port that the SMTP server is listening to.

Figure 4–9 Web Tier Configuration Screen



Click **Next** to continue.

4.3.2.12 Virtual Hosts Configuration Screen

This screen, shown in Figure 4–10, contains the configuration details for the domains on the virtual hosts.

Specify the following information for each application domain listed:

- **Internal Name** - the host name or IP address where the Webtier listens on the internal virtual host for this domain.
- **Internal Port** - port for this internal virtual host. Should be visible only from inside the firewall.
- **External Name** - the host name or IP address for the external virtual host for this product family or middleware dependency. The host:port should be visible from outside the firewall.
- **External Port** - port to be used for this external virtual host. The host:port should be visible from outside the firewall.

If you selected **Name-based** on the Web Tier Configuration screen, specify the following information for each domain listed:

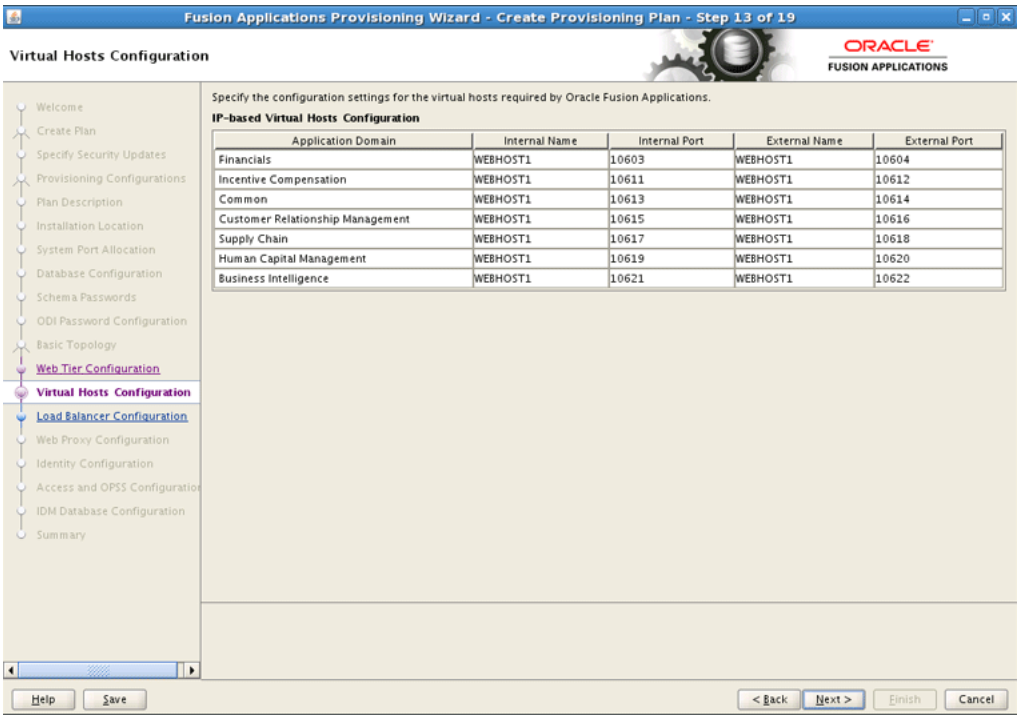
- **Internal.Name** - the DNS name for this internal virtual host. For example, for Financials, the name might be *fin-internal*.
- **External.Name** - the DNS name for this external virtual host. For example, for Financials, the name might be *fin*.

If you selected **Port-based** on the Web Tier Configuration screen, specify the following information for each domain listed:

- **Internal Port** - the port that is visible only from inside the firewall for this domain.

- **External Port** - the port that is visible from outside the firewall for this domain.

Figure 4–10 Virtual Hosts Configuration Screen



Click **Next** to continue.

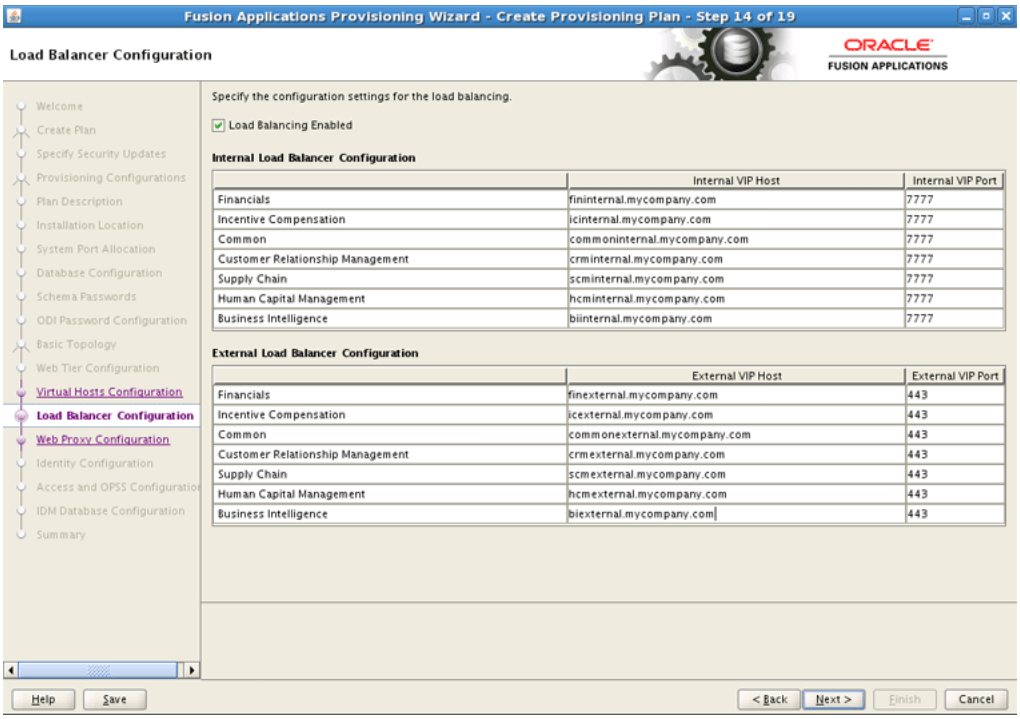
4.3.2.13 Load Balancer Configuration Screen

This screen, shown in Figure 4–11, enables you to distribute workload evenly across two or more hosts, network links, CPUs, hard drives, or other resources. Check **Load Balancing Enabled** to take advantage of this feature, and specify:

- **Internal Load Balancer Configuration** - the host and port for the internal Virtual IP (VIP).
- **External Load Balancer Configuration** - the host and port for external Virtual IP (VIP). It must have a publicly available address to be usable.

If you want to stop creating this plan and resume at a later date, click Save. This action creates a partial plan. A partial plan cannot be used to provision an environment.

Figure 4–11 Load Balancer Configuration Screen

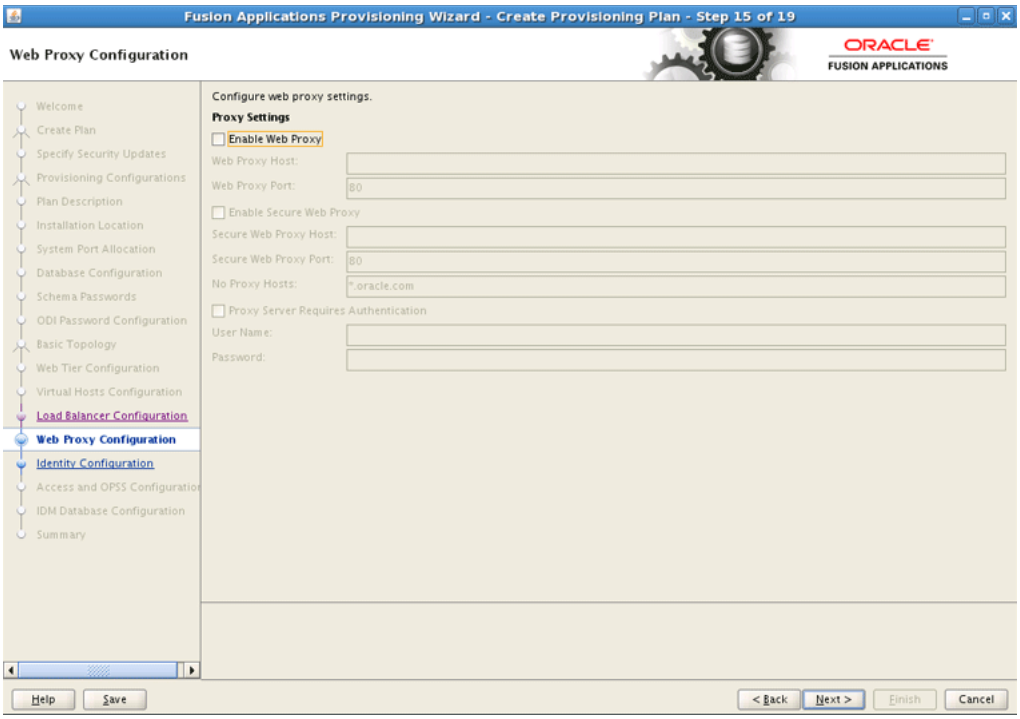


Click **Next** to continue.

4.3.2.14 Web Proxy Configuration Screen

This screen, shown in Figure 4–12, allows you to create Proxy Settings to enable users who want to use a proxy server to connect to the Internet.

Figure 4–12 Web Proxy Configuration Screen



4.3.2.15 Identity Management Configuration Screen

In these screens, shown in Figure 4–13 and Figure 4–14, enter the **Identity Management Configuration** parameters for the identity management infrastructure associated with this environment.

- **Super User Name** - enter the name of an existing user that should be granted administrator and functional setup privileges.
- **Enable seeding of Security Data** - controls the uploading of Oracle Fusion Applications security policies and user credentials into the Lightweight Directory Access Protocol (LDAP) as a part of provisioning. This option is selected by default. De-selecting it disables this action during provisioning.
- **Create Administrators Group** - indicate whether you created an "Administrators" group, whose members have specialized privileges for all Oracle Fusion Middleware components.
- **Create Monitors Group** - indicate whether you created a "Monitors" group, whose members have read-only administrative privileges to Oracle WebLogic domains.
- **Create Operators Group** - indicate whether you created an "Operators" group, whose members have Monitors privileges to Oracle WebLogic domains.
- **Identity Store Server Type** - indicate the type of identity store you set up: OID (Oracle Internet Directory) or OVD (Oracle Virtual Directory).
- **Identity Store Host** - enter the host or DNS name for your identity store LDAP service.
- **Identity Store Port** - port assigned to the identity store.
- **Identity Store Secure Port** - the SSL port for the identity store.
- **Identity Store User DN** - enter the Distinguished Name of the user you set up with read-write access to the LDAP.

- **Identity Store Password** - enter the password you set up for the user with read-write access to the LDAP.
- **Identity Store Read-Only User DN** - the Distinguished Name of the user with read-only access to the Identity Store LDAP.
- **Identity Store Read-Only Password** - enter the password you set up for the identity store read-only user.
- **Identity Store Enabled SSL** - select this option if your identity store is SSL-enabled and if the required certificates are provided in the IDM keystore file.
- **Identity Store User Name Attribute** - the type of user name attribute you configured in the identity store. Valid values are: user ID (uid), common name (CN), or email address.
- **Identity Store User Base DN** - enter the root Distinguished Name assigned to the upload of applications user data. This is the root for all the user data in your identity store.
- **Identity Store Group Base DN** - enter the root Distinguished Name for all the group data in your identity store.
- **OIM Admin Server Host** - enter the name of the host where the OIM Administration Server is installed.
- **OIM Admin Server Port** - the port where the OIM Administration Server listens.
- **OIM Administrator User Name DN** - enter the Distinguised Name you set up as the OIM administrator.
- **OIM Administrator Password** - enter the password you set up for the OIM administrator.
- **OIM Managed Server Host** - enter the virtual or real host name of the Oracle Identity Manager managed server where SPML callback and other OIM services are running.
- **OIM Managed Server Port** - enter the virtual or real port where the Oracle Identity Manager managed server listens.
- **OIM Endpoint Host** - enter the `http` termination address of Oracle Access Manager. Terminates at either a load balancer or the Oracle HTTP Server
- **OIM Endpoint Port** - the port where the endpoint host listens.
- **OIM Endpoint Enabled SSL** - select this option if the endpoint host is SSL-enabled.
- **IDM Keystore File** - enter the location of the JKS keystore containing the certificates for the Oracle Identity Management components. If SSL is not enabled, you can supply a "dummy" `.jks` file name. The file can be of any type, with or without content. Validation is only for the existence of the file.
- **IDM Keystore Password** - enter the password you set up for the IDM Keystore File. If you set up a "dummy" file, enter a "dummy" password. It can be of any format, as there is no validation other than for its existence.

Figure 4–13 Identity Management Configuration Screen (1)

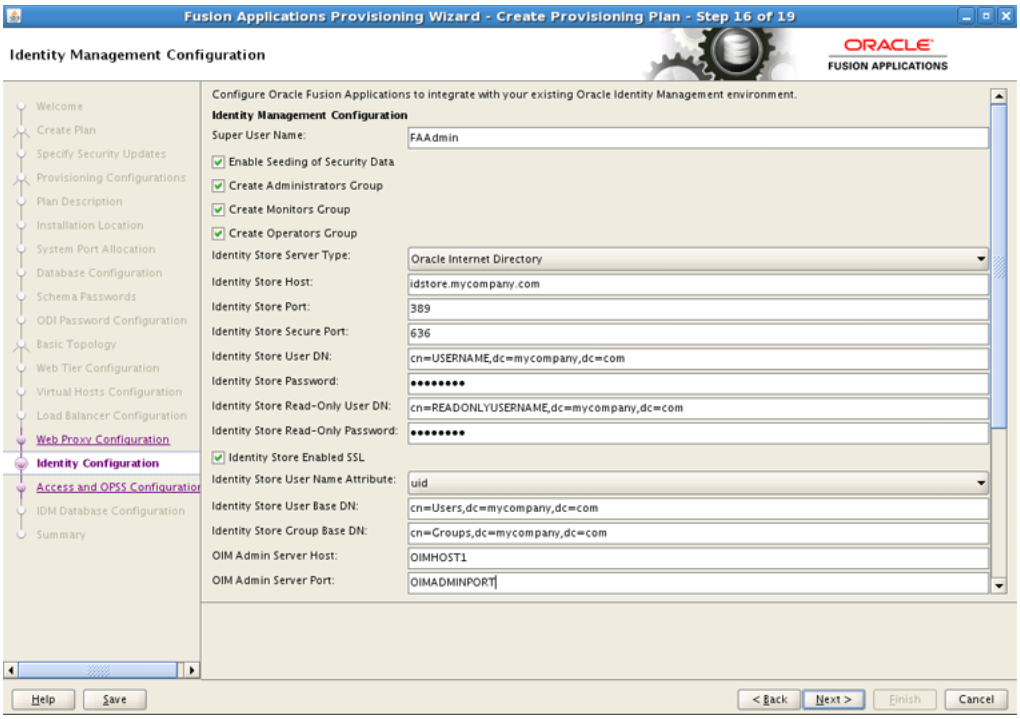
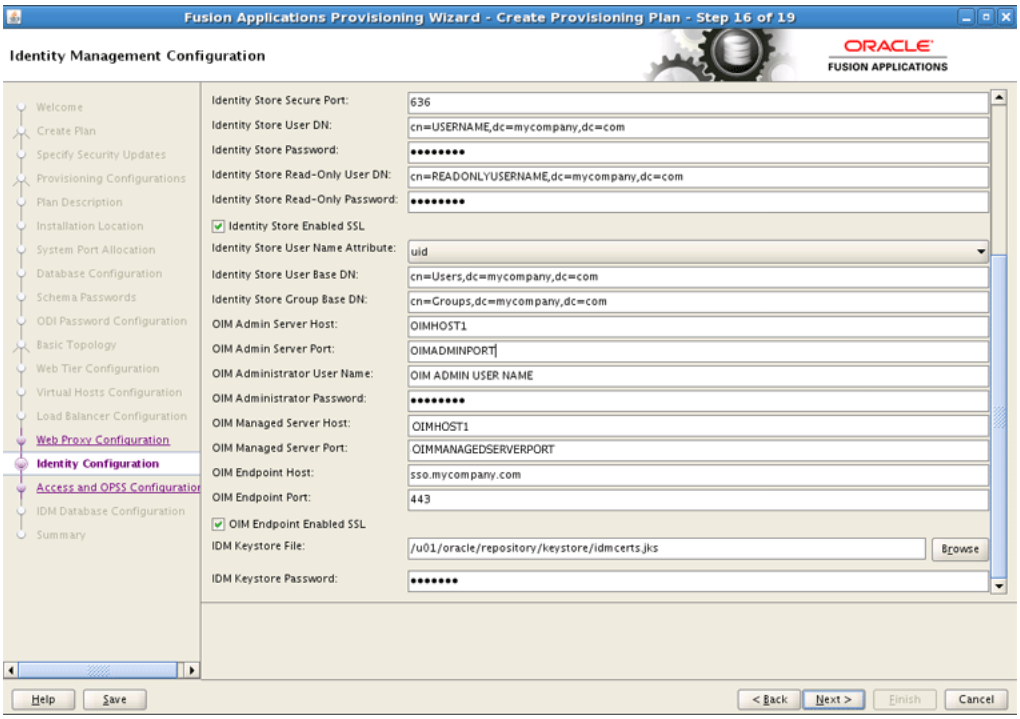


Figure 4–14 Identity Management Configuration Screen (2)



Click **Next** to continue.

4.3.2.16 Access and Policy Management Configuration Screen

Access and Policy Management Configuration provides identity administration and security functions such as Single Sign-On and policy management. In these screens, shown in Figure 4–15 and Figure 4–16, supply the following parameters to integrate with your existing Oracle Identity Management environment:

- **Oracle Access Manager Host** - enter the name of the host where the Oracle Access Manager is installed.
- **Oracle Access Manager Port** - port number for the Oracle Access Manager listener.
- **Access Server Identifier** - name used to identify the Oracle Access Server.
- **Enable Second Primary Oracle Access Manager** - select if you want to name a second Primary Oracle Access Manager for high availability.
- **Second Access Server Identifier** - enter the name of the second Primary Oracle Access Manager Server.

> **Note:** After connecting to the primary access server, provisioning is able to get the secondary access server connection information.

- **OAM Security Mode** - enter the OAM transport security mode you set up for this access server. Values are SIMPLE or OPEN. Enter the value you used when you set up when you installed OAM.
- **OAM Simple Mode Passphrase** - enter the passphrase you set up to secure the communication with the OAM Server. Required only if the mode is specified as SIMPLE.
- **OAM Policy Store Host** - enter the host name for OID where Oracle Platform Security Services (OPSS) policies are to be seeded.
- **OAM Policy Store Port** - number of the OID port for OPSS policy store.
- **OAM Policy Store Secure Port** - secure port for the LDAP where OAM policies are to be seeded.
- **OAM Policy Store Read-Write User Name** - enter the Distinguished Name of the user that you set up with write privileges to the OPSS policy store.
- **OAM Policy Store Password** - enter the password you set up for the OPSS policy store user with read-write privileges.
- **OPSS Policy Store JPS Root Node** - enter Distinguished Name of the node you set up to be used as the OPSS policy root.
- **Create OPSS Policy Store JPS Root Node** - enabled only if the **Enable Seeding of Security Data** checkbox on the Identity Management Configuration screen is selected. Select this option if you want to create the OPSS JPS Root Node.
- **OAM Policy Store SSL Enabled** - select this option if the OID used for storing OPSS policies is SSL-enabled and the required certificates are provided in the IDM keystore.
- **Webgate Password/Confirm Password** - specify a password for the Resource WebGate. Re-type to confirm the password. If seeding of security data is disabled, the password must be the existing WebGate password.

Figure 4–15 Access and Policy Management Configuration Screen (1)

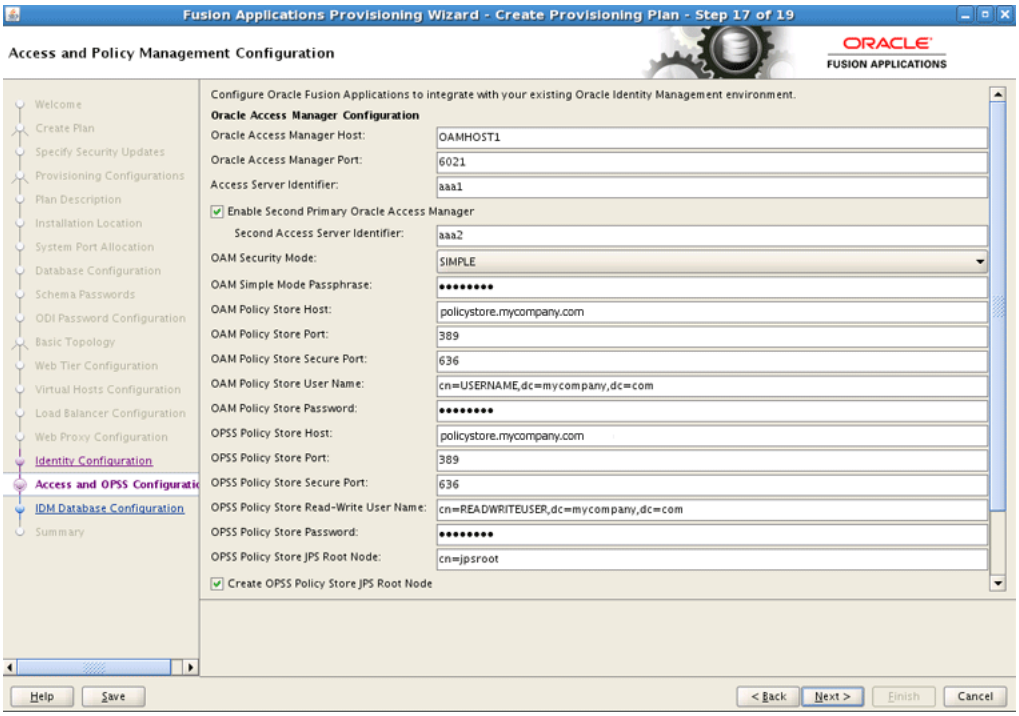
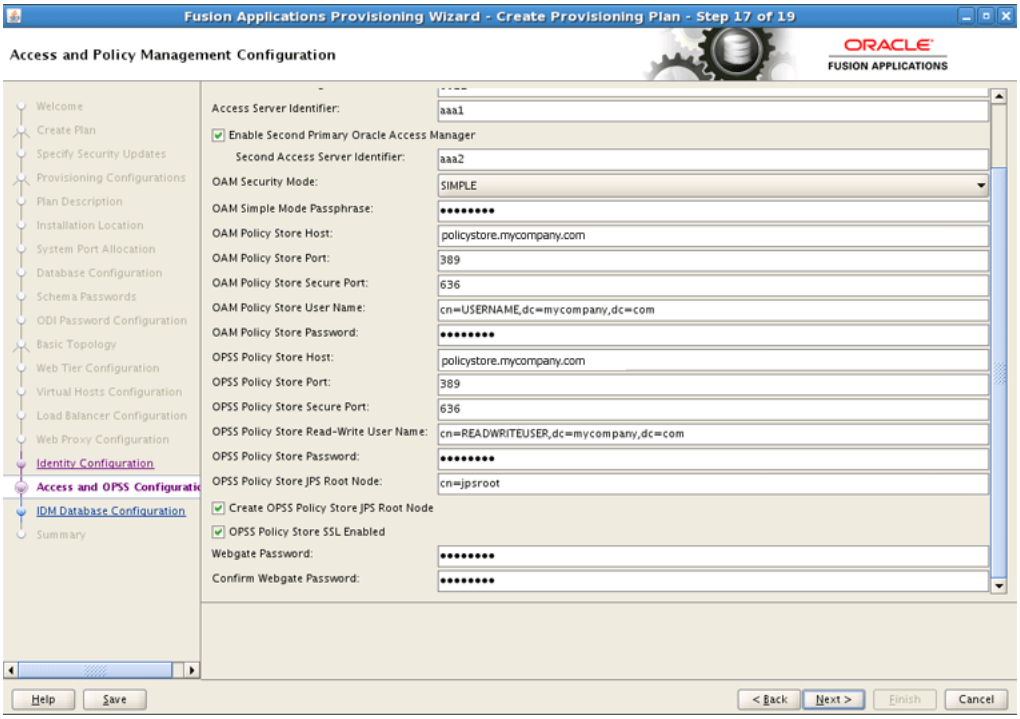


Figure 4–16 Access and Policy Management Configuration Screen (2)



Click **Next** to continue.

4.3.2.17 IDM Configuration Screen

In this screen, shown in Figure 4–17, enter the configuration details you specified when you installed the database for the Oracle Identity Manager (OIM).

Select **Real Application Clusters Database** if you have installed an OIM database based on Oracle Real Application Clusters (Oracle RAC). Specify the **Service Name**.

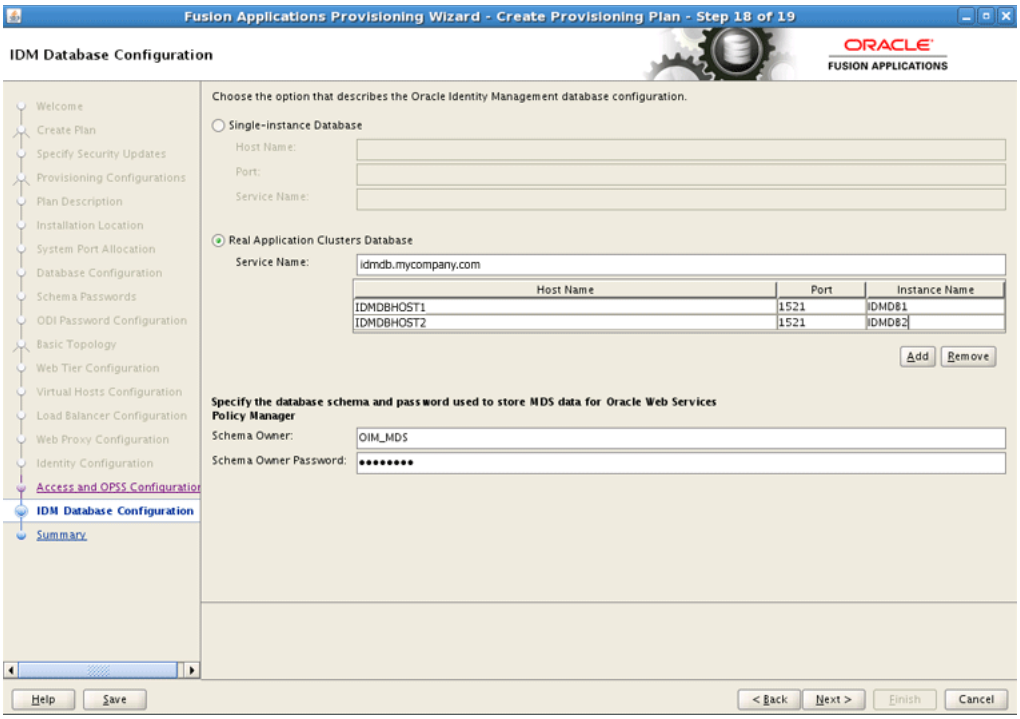
To identify the Oracle RAC instances, click **Add** to create a new row in the table. To delete a row, select it and click **Remove**. Enter the following information for each instance:

- **Host Name** - the name of the Oracle RAC host where you have installed the OIM database. In this field, you select an existing host or enter a new one. As you enter values for a new host, the list of hosts is populated with the new information.
- **Port** - listening port of the RDBMS.
- **Instance Name** - the Oracle RAC database instance name

Specify the database schema and password used to store the Metadata Service (MDS) Repository data for Oracle Web Services Policy Manager.

- **Schema Owner** - the MDS schema in the OIM database that is used by Oracle Web Services Policy Manager.
- **Schema Owner Password** - the password for the MDS schema.

Figure 4–17 IDM Configuration Screen



Click **Next** to continue.

4.3.2.18 Summary Screen

Review the information on this screen. If it is not what you expected or intended, click **Back** to return to the interview flow screen that needs to be changed, or click the name of the screen in the left navigation pane.

Descriptive information for this plan (if any) and database connection details are displayed under **Global Settings**. Each ***product family*** **Domain** to be created is listed along with the configuration details you have previously entered.

If you are satisfied with the information as displayed, specify the following information:

- **Provisioning Plan Name** - the executable file that contains the configuration details of this provisioning plan.
- **Provisioning Summary** - a text document that summarizes the details of this provisioning plan. You cannot use this file to execute the plan.
- **Directory** - the directory path to the location where you save the plan and the summary document.

Make a note of the name and location where you saved the executable file. You must supply this information to the Installation Wizard for other options.

Click **Finish** to save the `provisioning.plan` and `provisioning.summary` files to *`ORACLE_BASE`*`/repository/provisioning/bin`.

4.3.3 Running the Provisioning Commands to Install Components

The provisioning commands that install components perform the following tasks:

- Set up WEBHOST1
- Set up the debug flag
- Run the pre-verify phase
- Run the installation phase

Prerequisites for running the provisioning commands

Before running the provisioning commands which run the ant targets (preverify, install, and so on), do the following:

- Check the release notes for any known workarounds.
- Ensure that the preverify target is passed before you move on to other targets.
- Ensure that all commands with ant targets say "Build Successful" when they pass.
- Set the *`JAVA_HOME`* variable to *`ORACLE_HOME`*`/repository/jdk6`.
- Create the `/etc/oraInst.loc` file with the following entries:

  ```
  inventory_loc=ORACLE_HOME/oraInventory

  inst_group=usergroup
  ```

- For commands that fail, use the following location to debug:

  `ORACLE_BASE``/products/logs/provisioning/`*`CRMHOST1`*`/runProvisioning-`*`targetname`*`.log`

Note: If for any reason you need to run any target again, run the following:

```
./runProvisioning.sh -plan ./provisioning.plan -override
./overrides.properties -target cleanup-targetname

./runProvisioning.sh -plan ./provisioning.plan -override
./overrides.properties -target restore-targetname
```

Then run the target again.

Task 1 Set up *WEBHOST1*

For this task, *WEBHOST1* is the host you configured in Figure 4–9, "Web Tier Configuration Screen". *WEBHOST1* and *CRMHOST1* do not have a common shared storage. Copy the full repository from *CRMHOST1* to *WEBHOST1*.

In addition, copy the `provisioning.plan` and `overrides.properties` files to *ORACLE_HOME*`/repository/provisioning/bin`.

Be sure to maintain the same directory structure on *WEBHOST1*.

Task 2 Set up the debug flag

Run the following commands:

- **On *CRMHOST1*:** `export DEBUG_PROVISIONNING=true`
- **On *WEBHOST1*:** `export DEBUG_PROVISIONNING=true`

Task 3 Run the pre-verify phase

Run the following commands from *ORACLE_HOME*`/repository/provisioning/bin`:

- **On *CRMHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target preverify`
- **On *WEBHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target preverify`

Task 4 Run the installation phase

Run the following commands from *ORACLE_HOME*`/respository/provisioning/bin`:

- **On *CRMHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target install`
- **On *WEBHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target install`

Note: If the relative path for the plan (`./provisioning.plan`) and override properties (`./override.properties`) does not work, give a fully qualified path instead.

4.4 Configuring Components

The provisioning commands that configure components perform the following tasks:

- Run the pre-configure phase

- Run the configure phase
- Run the configure secondary phase
- Run the post-configure phase
- Run the start-up phase
- Run the validate phase
- Configure email servers

Note: If for any reason you have to run the configure stage again, Provisioning allows you to do that. Clean up and restore targets need to be run. For example, if you need to run configure again, run the following:

```
./runProvisioning.sh -plan ./provisioning.plan -override
./overrides.properties -target cleanup-configure

./runProvisioning.sh -plan ./provisioning.plan -override
./overrides.properties -target restore-configure
```

Once the cleanup and restore builds are successful, you can run the configure target again.

This applies to all tasks related to configuring components.

Note: If the relative path for the plan (`./provisioning.plan`) and override properties (`./override.properties`) does not work, give a fully qualified path instead.

Task 1 Run the pre-configure phase

Before running the pre-configure phase, copy the *ORACLE_BASE*`/products/webtier_dmz_artifacts.zip` file from the *CRMHOST1* non-DMZ computers to *WEBHOST1 ORACLE_BASE*.

Run the following commands from *ORACLE_HOME*`/repository/provisioning/bin`:

- **On *CRMHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target preconfigure`
- **On *WEBHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target preconfigure`

If the pre-configure target fails because of the shared directory specified in the RCU as the Oracle Business Intelligence Enterprise Edition (OBIEE) backup directory, do the following:

1. Log in to a RAC instance using **sqlplus** as the `FUSION_BIPLATFORM` user.
2. Specify an Oracle Automatic Storage Management (Oracle ASM) directory to be used as the OBIEE backup directory using a command similar to the following:

```
SQL> create or replace directory FUSIONAPPS_PROV_RECOVERY_DIR as
'+DATA/crmdb/datafile';
```

```
SQL> commit;
```

Task 2 Run the configure phase

Run the following commands from *ORACLE_HOME*/repository/provisioning/bin:

- **On *CRMHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target configure`
- **On *WEBHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target configure`

Task 3 Run the configure secondary phase

Run the following commands from *ORACLE_HOME*/repository/provisioning/bin:

- **On *CRMHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target configure-secondary`
- **On *WEBHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target configure-secondary`

Task 4 Run the post-configure phase

Run the following commands from *ORACLE_HOME*/repository/provisioning/bin:

- **On *CRMHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target postconfigure`
- **On *WEBHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target postconfigure`

Task 5 Run the start-up phase

Run the following commands from *ORACLE_HOME*/repository/provisioning/bin:

- **On *CRMHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target startup`
- **On *WEBHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target startup`

Task 6 Run the validate phase

Run the following commands from *ORACLE_HOME*/repository/provisioning/bin:

- **On *CRMHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target validate`
- **On *WEBHOST1*:** `./runProvisioning.sh -plan ./provisioning.plan -override ./overrides.properties -target validate`

Once all the scripts have run successfully, CRM provisioning is complete. For information about the resulting directory structure, see Section 2.4.1, "Directory Structure."

For information about the tasks these scripts perform, see *Oracle Fusion Applications Installation Guide*.

Task 7 Configure email servers

To configure an email server as a delivery channel to be used with Oracle Business Intelligence Publisher, see "Adding an E-mail Server" in the chapter "Setting Up Delivery Destinations" in *Oracle Fusion Middleware Administrator's Guide for Oracle Business Intelligence Publisher*.

4.5 Performing Post-Provisioning Validation

After provisioning, access the following URLs, ensuring that the Administration console is visible:

- `http://crminternal.mycompany.com:7777/console`
- `http://hcminternal.mycompany.com:7777/console`
- `http://scminternal.mycompany.com:7777/console`
- `http://commoninternal.mycompany.com:7777/console`
- `http://fininternal.mycompany.com:7777/console`
- `http://biinternal.mycompany.com:7777/console`
- `http://icinternal.mycompany.com:7777/console`

For the following URLs, ensure that the Oracle Fusion Applications login screen is visible.

- `https://crmexternal.mycompany.com/sales/faces/mooOpportunityHome`
- `https://crmexternal.mycompany.com/crmPerformance/faces/TerritoriesMain`
- `https://crmexternal.mycompany.com/contractManagement/faces/ContractsDashboard`
- `https://crmexternal.mycompany.com/customer/faces/CustomerCtrWorkarea`
- `https://crmexternal.mycompany.com/marketing/faces/LeadsDashboard`
- `https://crmexternal.mycompany.com/orderCapture/faces/SalesCatalogAdmin`
- `https://commonexternal.mycompany.com/helpPortal/faces/AtkHelpPortalMain`
- `https://commonexternal.mycompany.com/homePage/faces/AtkHomePageWelcome`
- `https://biexternal.mycompany.com/analytics`

5

Scaling Out Oracle HTTP Server

This chapter assumes that configuration of the first Oracle HTTP Server was completed during the provisioning process.

This chapter includes the following topics:

- Section 5.1, "Performing the Scaleout"
- Section 5.2, "Installing WebGate Patches"
- Section 5.3, "Wiring Oracle HTTP Server with Load Balancer"
- Section 5.4, "Validating Oracle HTTP Server on WEBHOST2"

5.1 Performing the Scaleout

To scale out Oracle HTTP Server:

1. Reboot *WEBHOST2* to start the scaleout from a clean machine.
2. Create the directory *ORACLE_BASE*`/repository/installers` with the same user that installed Oracle HTTP Server on *WEBHOST1*.
3. Copy or `ftp` the installers from *WEBHOST1* to *WEBHOST2*:

   WEBHOST1`> `*ORACLE_BASE*`/repository/installers/webgate`

   to

   WEBHOST2`> `*ORACLE_BASE*`/repository/installers`

   WEBHOST1`> `*ORACLE_BASE*`/repository/installers/webtier`

   to

   WEBHOST2`> `*ORACLE_BASE*`/repository/installers`

   WEBHOST1`> `*ORACLE_BASE*`/repository/installers/webtier_patchset`

   to

   WEBHOST2`> `*ORACLE_BASE*`/repository/installers`

4. Run the following command to install Oracle Web Tier on *WEBHOST2*:

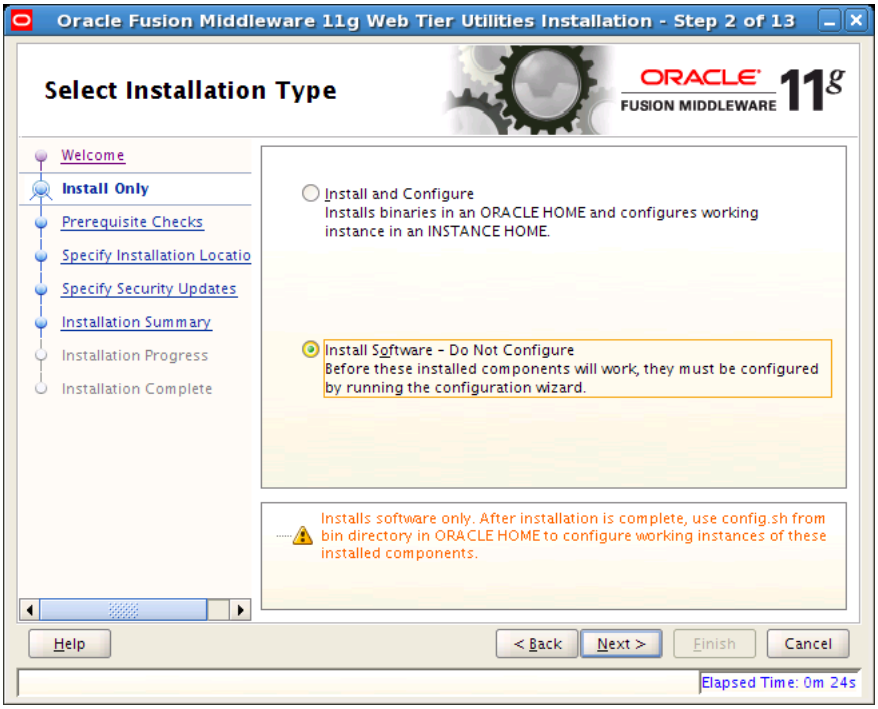
   ORACLE_BASE`/repository/installers/webtier/Disk1/runInstaller`

   The Oracle Fusion Middleware 11*g* Oracle Web Tier Utilities Configuration Welcome window opens.

5. Click **Next** to start the installation.

   The Select Installation Type window, shown in Figure 5–1, opens.

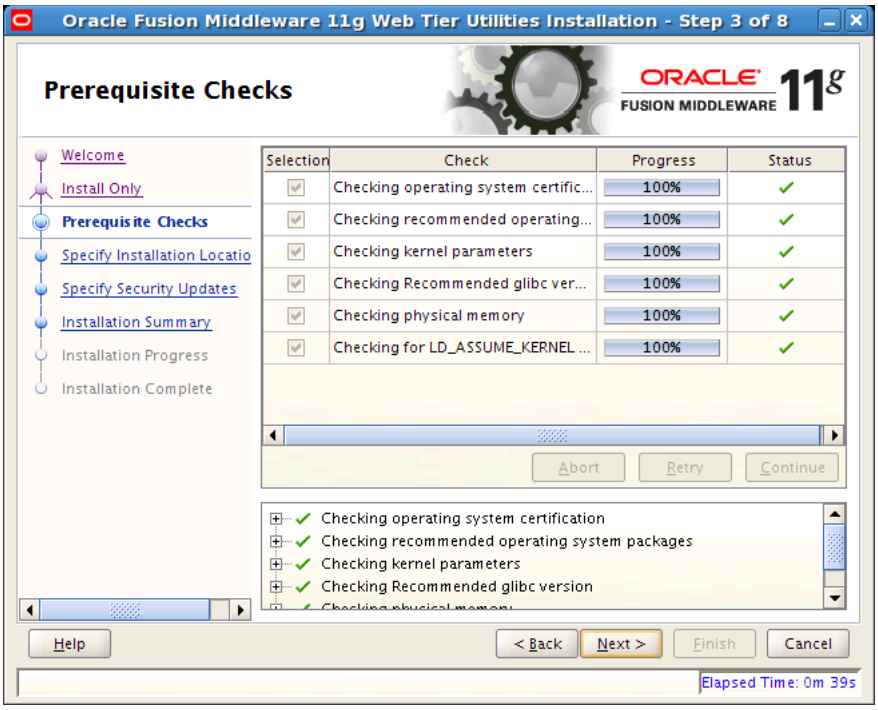
Figure 5–1 Select Installation Type Window



6. **Select Install Software - Do Not Configure** and click **Next**.

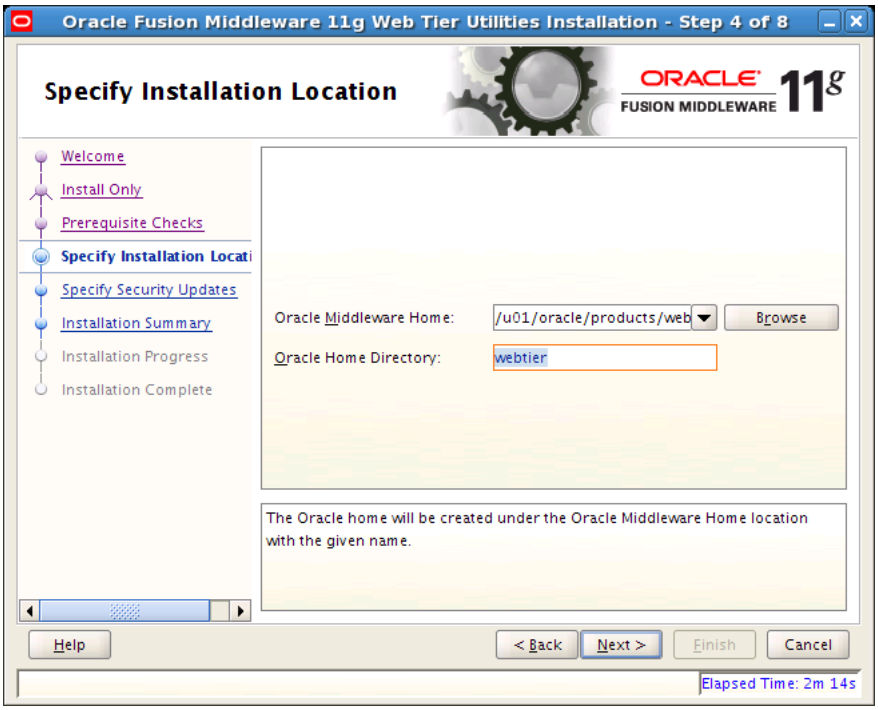
   The Prerequisite Checks window, shown in Figure 5–2, opens.

Figure 5–2 Prerequisite Checks Window



Click **Next**. The Specify Installation Location window, shown in Figure 5–3, opens.

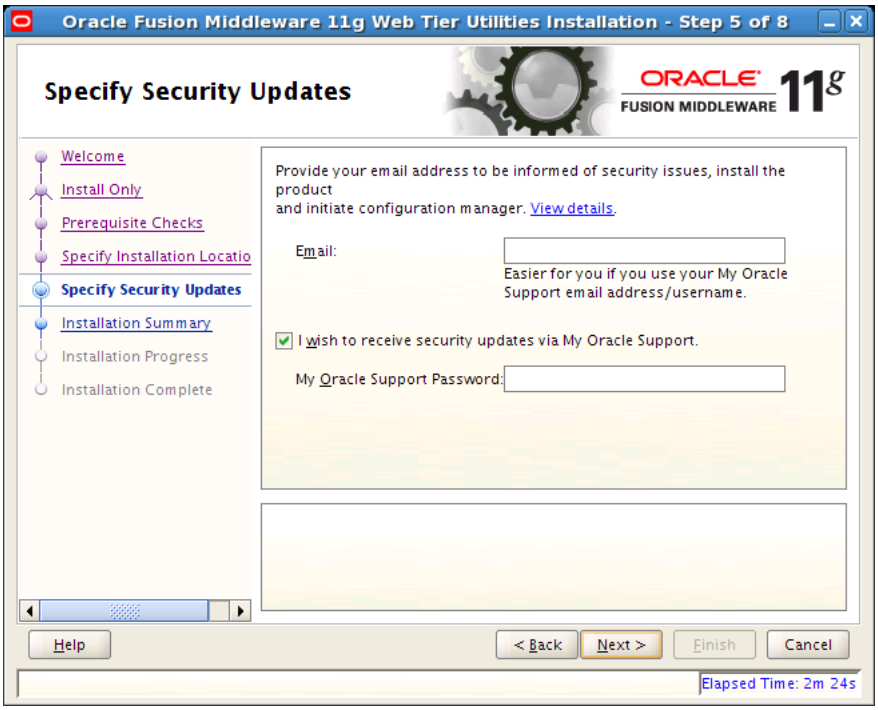
Figure 5–3 Specify Installation Location Window



Select the path to Oracle Middleware Home and enter a name for the home directory. For example, `/u01/oracle/products/webtier_mwhome/`. Click **Next**.

7. In the Specify Security Updates window, shown in Figure 5–4, do the following:
   - Enter an email address
   - Indicate that you wish to receive security updates from My Oracle Support
   - Enter your My Oracle Support password

Figure 5–4 Specify Security Updates Window

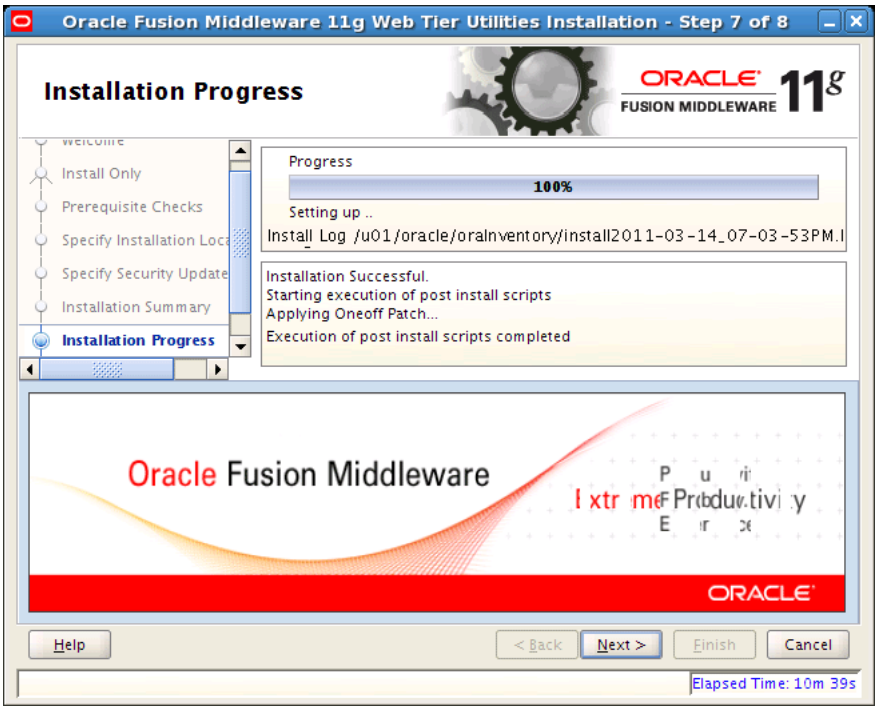


Click **Next**. The Installation Summary window, shown in Figure 5–5, opens.

Figure 5–5 Installation Summary Window

8. Click **Install**. The Installation Progress window, shown in Figure 5–6, opens.

Figure 5–6 Installation Progress Window



Click **Next** when the installation has finished. The Installation Complete window, shown in Figure 5–7, opens.

Figure 5–7 Installation Complete Window

Click **Finish**.

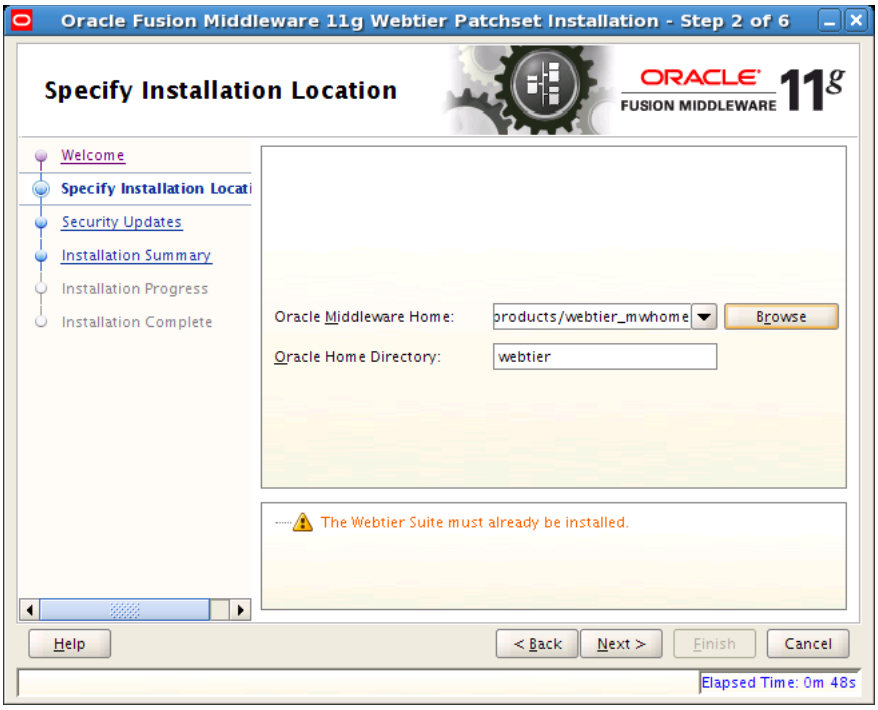
9. Run the following command to install the Oracle Web Tier patchset on *WEBHOST2*:

   ORACLE_BASE/repository/installers/webtier_patchset/Disk1/runInstaller

   The Oracle Fusion Middleware 11*g* Webtier Patchset Installation Welcome window displays. Click **Next** to being the installation.

   The Specify Installation Location window, shown in Figure 5–8, opens.

Figure 5–8 Specify Installation Location Window (Patchset)

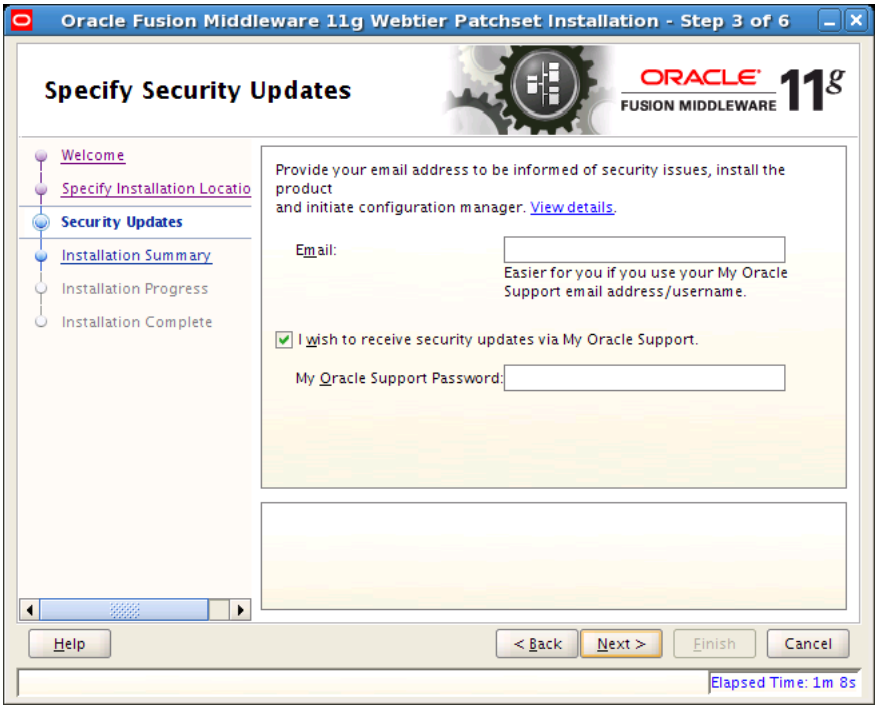


10. Select the path to Oracle Middleware Home and enter a name for the home directory. For example, `/u01/oracle/products/webtier_mwhome/`. Click **Next**.

In the Specify Security Updates window, shown in Figure 5–9, do the following:

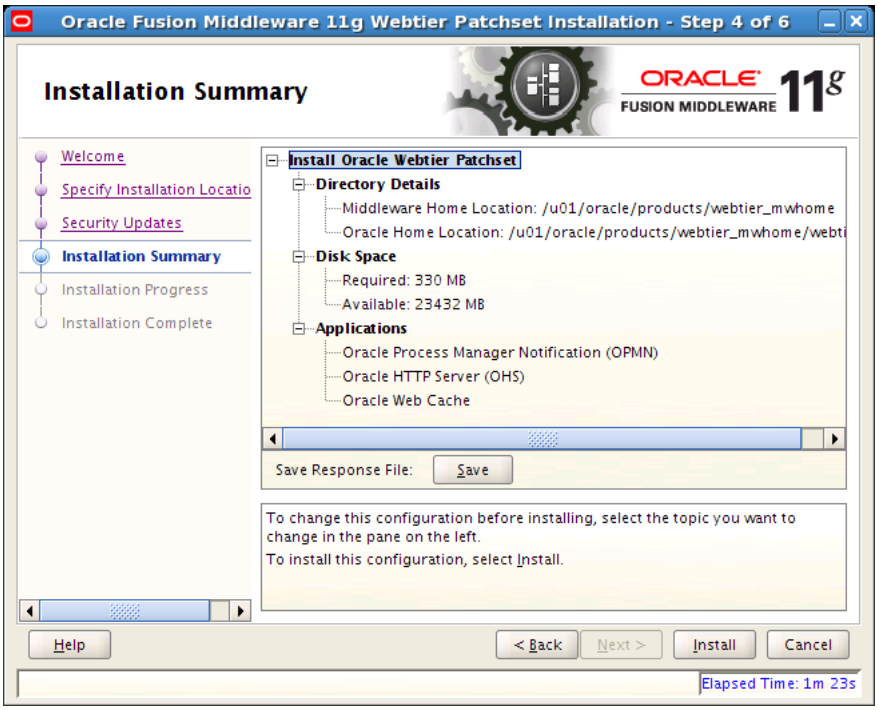
- Enter an email address
- Indicate that you wish to receive security updates from My Oracle Support
- Enter your My Oracle Support password

Figure 5–9 Specify Security Updates Window (Patchset)



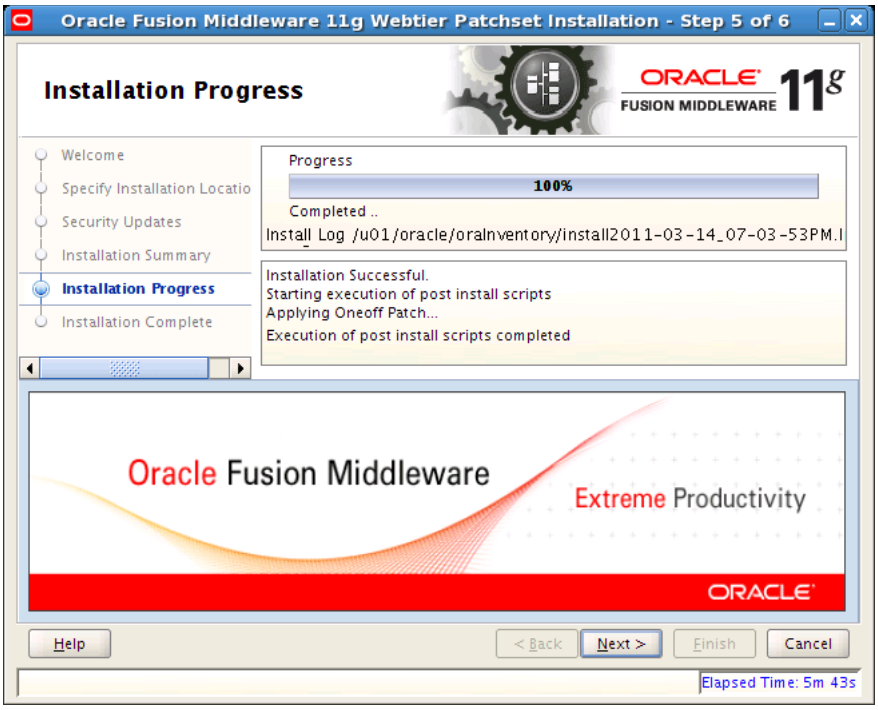
11. Click **Next**. The Installation Summary Window, shown in Figure 5–10, opens.

Figure 5–10 Installation Summary Window (Patchset)



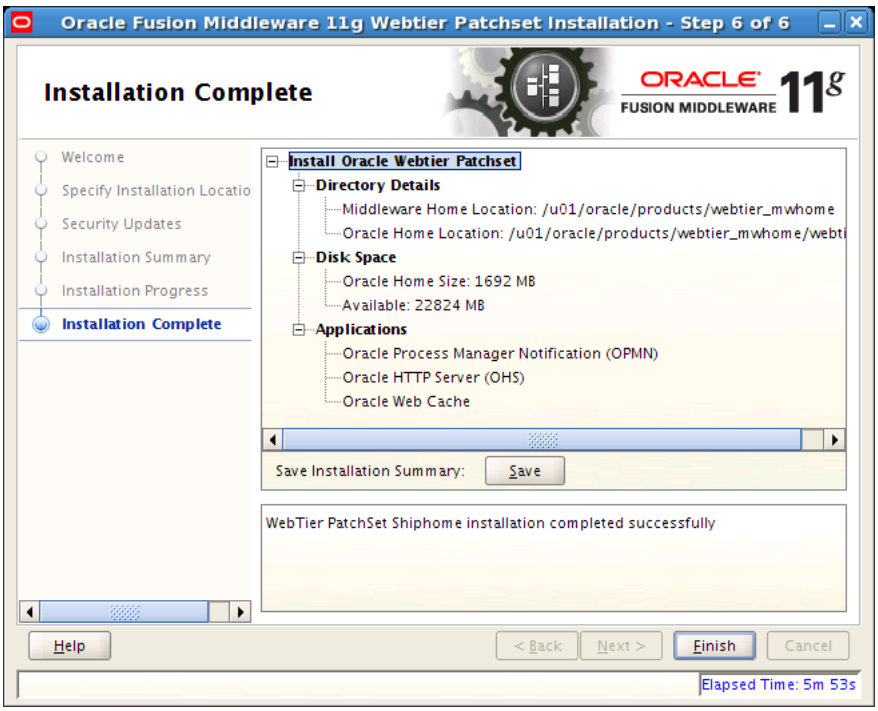
12. Click **Install**. The Installation Progress window, shown in Figure 5–11, opens.

Figure 5–11 Installation Progress Window (Patchset)



13. Click **Next** when the installation has finished. The Installation Complete window, shown in Figure 5–12, opens.

Figure 5–12 Installation Complete Window (Patchset)



14. Click **Finish**.

15. View the directory structure that has been created:

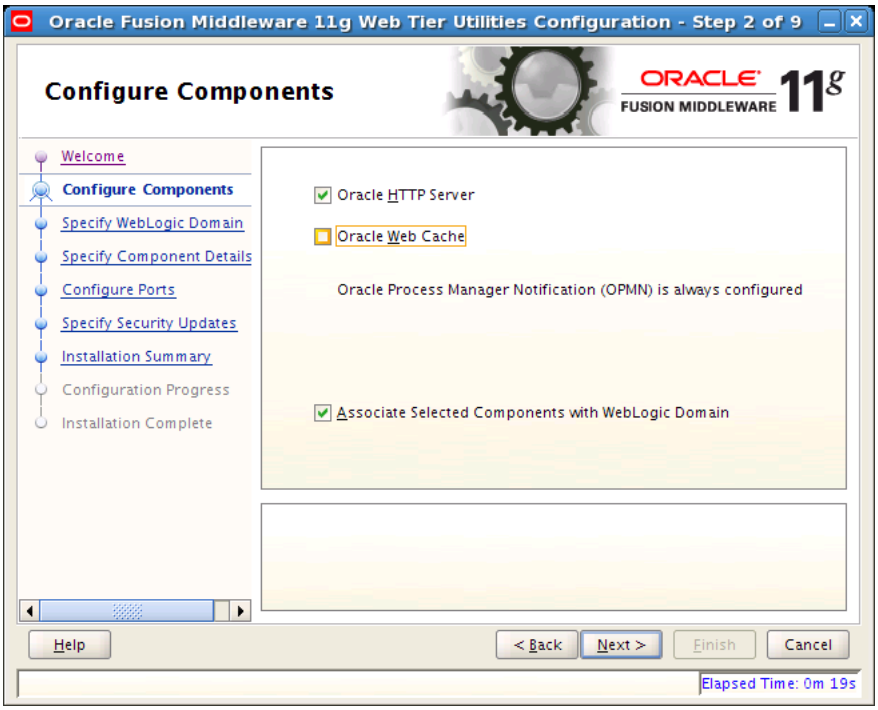
```
cd ORACLE_BASE/products/webtier_mwhome
```

16. Begin configuring the Oracle Web Tier components:

```
cd ORACLE_BASE/products/webtier_mwhome/webtier/bin

run ./config.sh
```

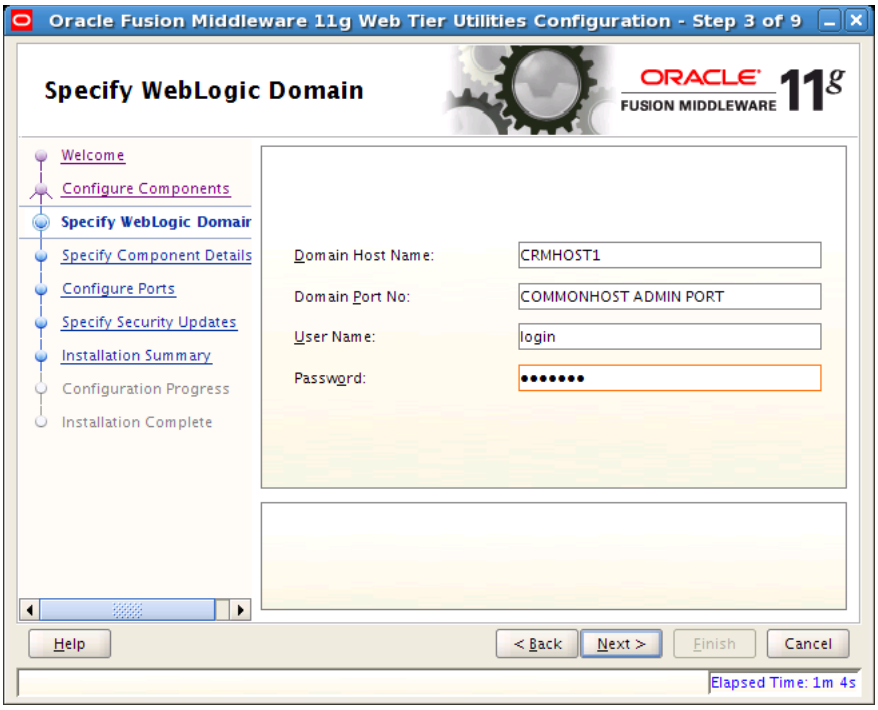
The Configure Components window, shown in Figure 5–13, opens.

Figure 5–13 Configure Components Window



17. Select **Oracle HTTP Server** and **Associate Selected Components with WebLogic Domain**.
18. Click **Next**. The Specify WebLogic Domain window, shown in Figure 5–14, opens.

Figure 5–14 Specify WebLogic Domain Window

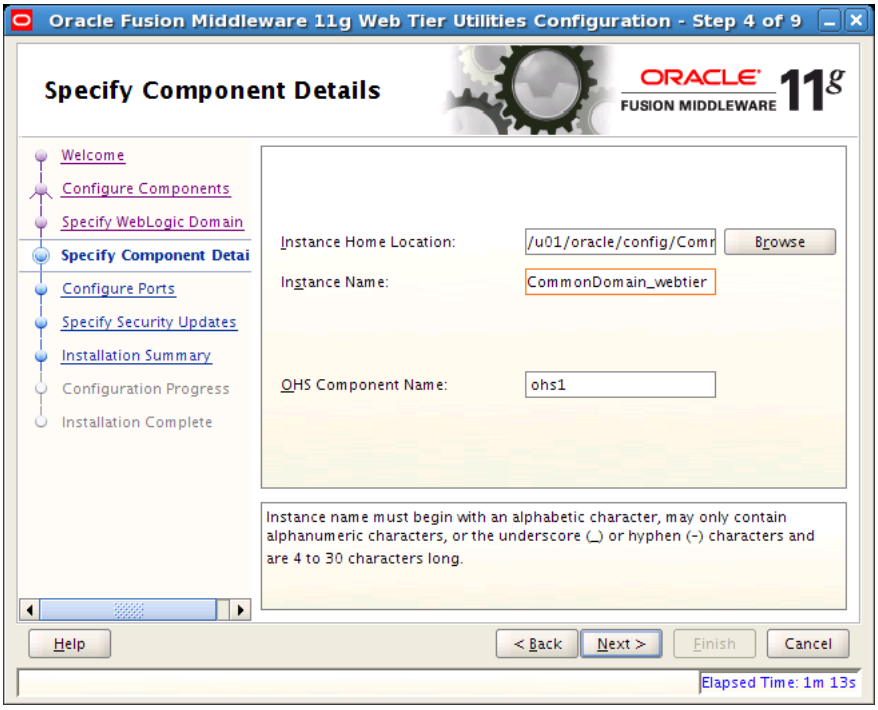


19. Enter the following:
    - a domain host name; for example, `CRMHOST1`
    - a domain port number; for example, `COMMONHOST ADMIN PORT`
    - your CommonDomain Administration Server user name
    - your CommonDomainAdministration Server password

    Note: Associate Oracle HTTP Server with the CommonDomain that Provisioning installed in Chapter 4.

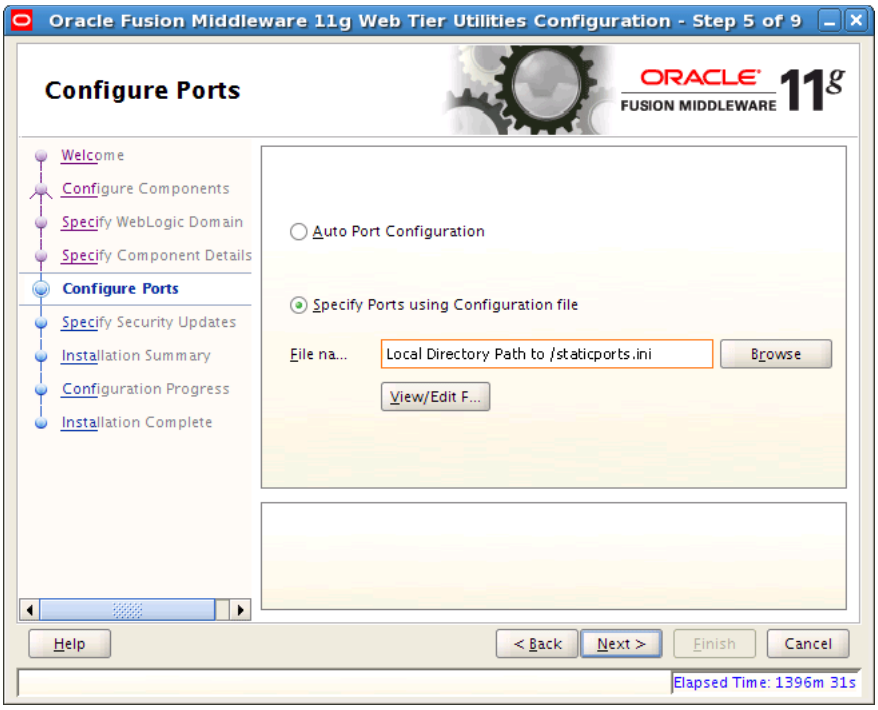
20. Click **Next**. The Specify Component Details window, shown in Figure 5–15, opens.

Figure 5–15 Specify Component Details Window



21. Do the following:
    - Select the instance home location; for example, `/u01/oracle/config/CommonDomain_webtier1`
    - Enter the instance name; for example, `CommonDomain_webtier1`
    - Enter the Oracle HTTP Server component name; for example `ohs1`
22. Click **Next**. The Configure Ports window, shown in Figure 5–16, opens.

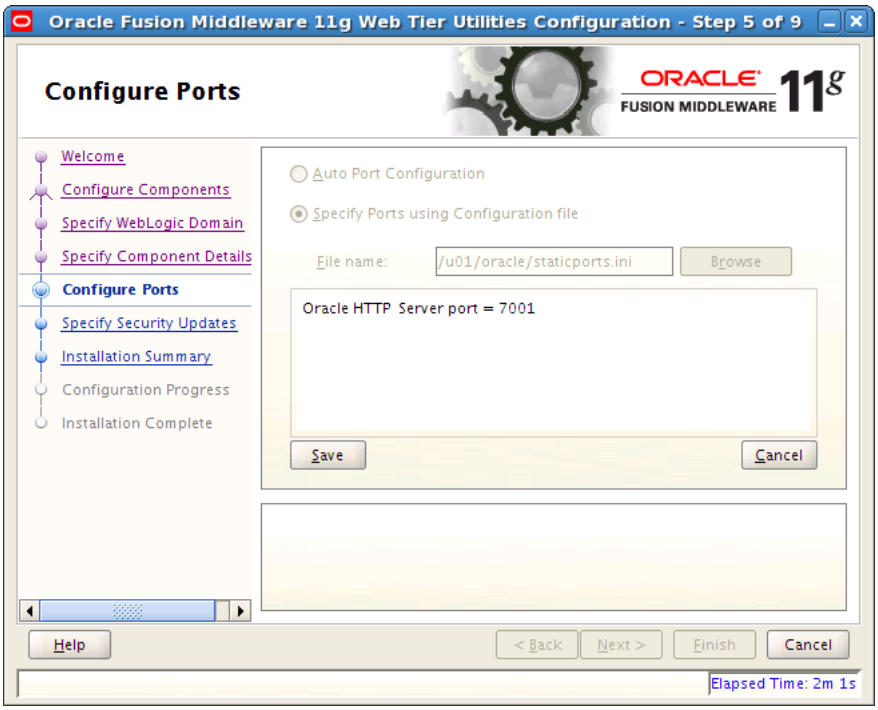
Figure 5–16 Configure Ports Window (1)



23. Select **Specify Ports using Configuration file** and click **View/Edit**.

In the text field that displays, shown in Figure 5–17, enter `Oracle HTTP Server port = 7001`.

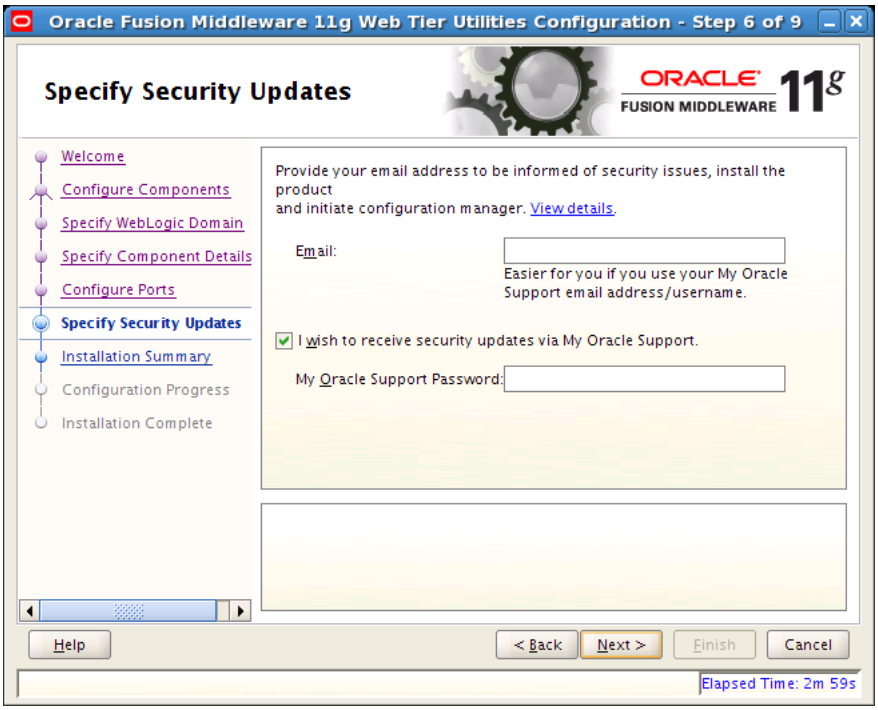
Figure 5–17 Configure Ports Window (2)



Click **Save** and then click **Next**.

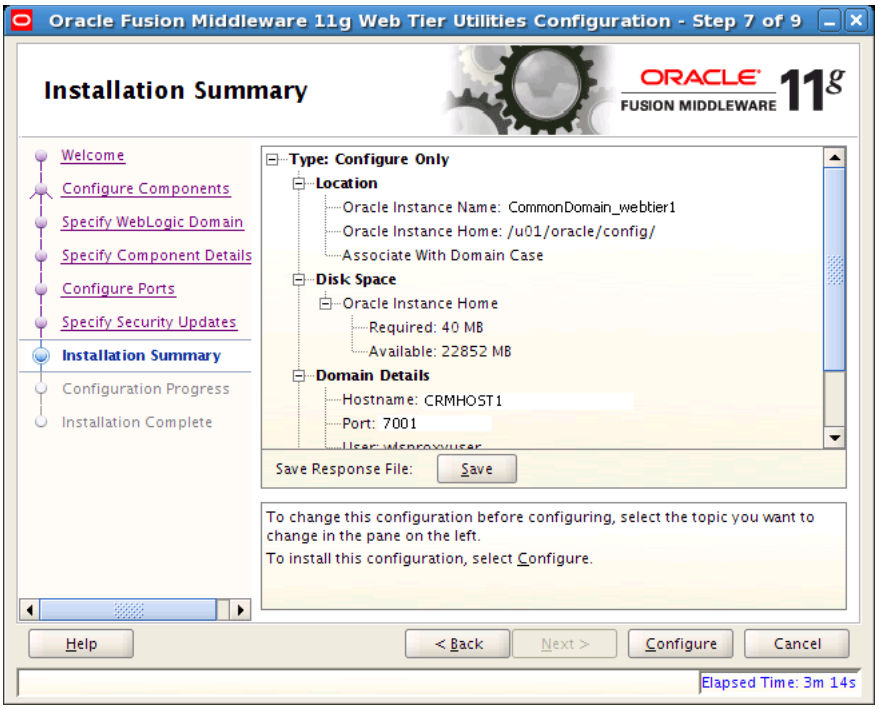
The Specify Security Updates window, shown in Figure 5–18, opens.

Figure 5–18 Specify Security Updates



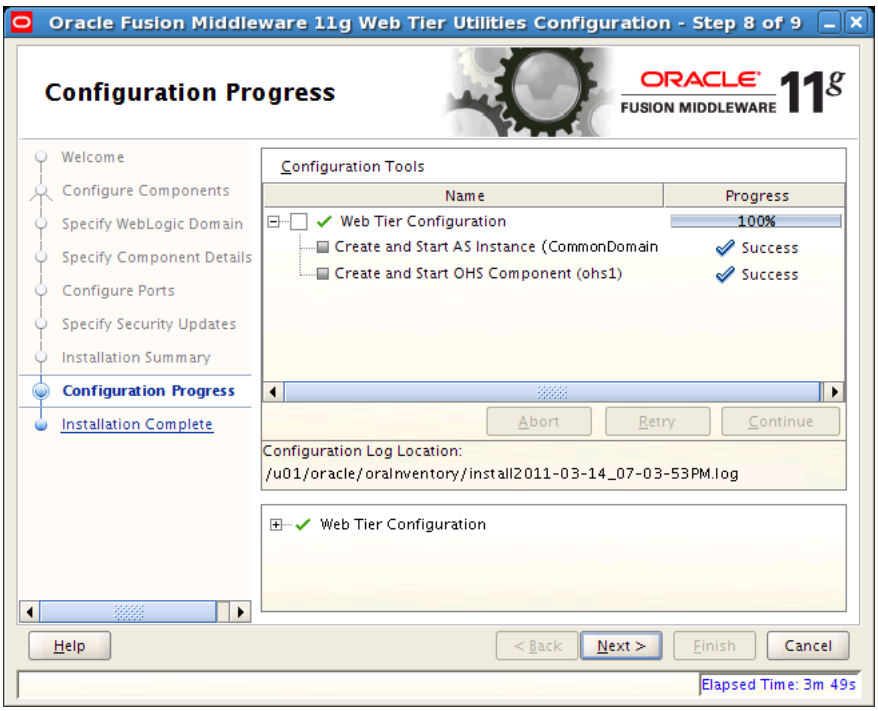
24. Do the following:
    - Enter an email address
    - Indicate that you wish to receive security updates from My Oracle Support
    - Enter your My Oracle Support password
25. Click **Next**. The Installation Summary window, shown in Figure 5–19, opens.

Figure 5–19 Installation Summary Window



26. Click **Configure** to install the configuration. The Configuration Progress window, shown in Figure 5–20, opens.

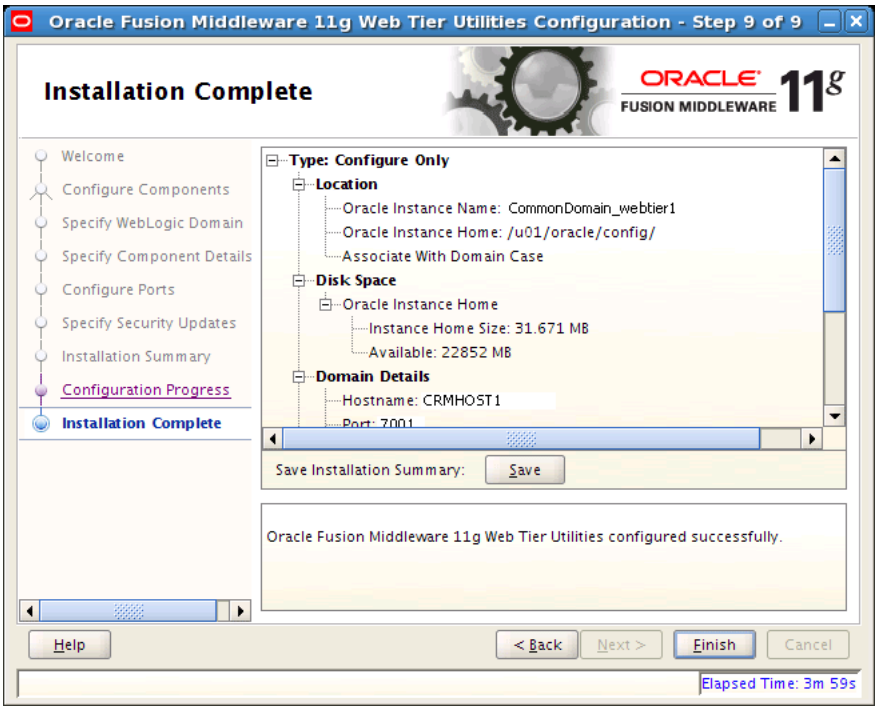
Figure 5–20 Configuration Progress Window



27. Click **Next**.

When the installation completes, the Installation Complete window, shown in Figure 5–21, opens.

Figure 5–21 Installation Complete Window



28. Click **Finish**.
29. Copy or `ftp` the FusionVirtualHost files from *`WEBHOST1`* to *`WEBHOST2`*:

    `WEBHOST1``> `*`ORACLE_BASE`*`/config/CommonDomain_webtier1/config/OHS/ohs1/moduleconf`

    to

    `WEBHOST2``> `*`ORACLE_BASE`*`/config/CommonDomain_webtier1/config/OHS/ohs1/moduleconf/`

30. After the copy, change all *`WEBHOST1`* entries in the `.conf` files to *`WEBHOST2`*.
31. Restart the Oracle HTTP Server instance:

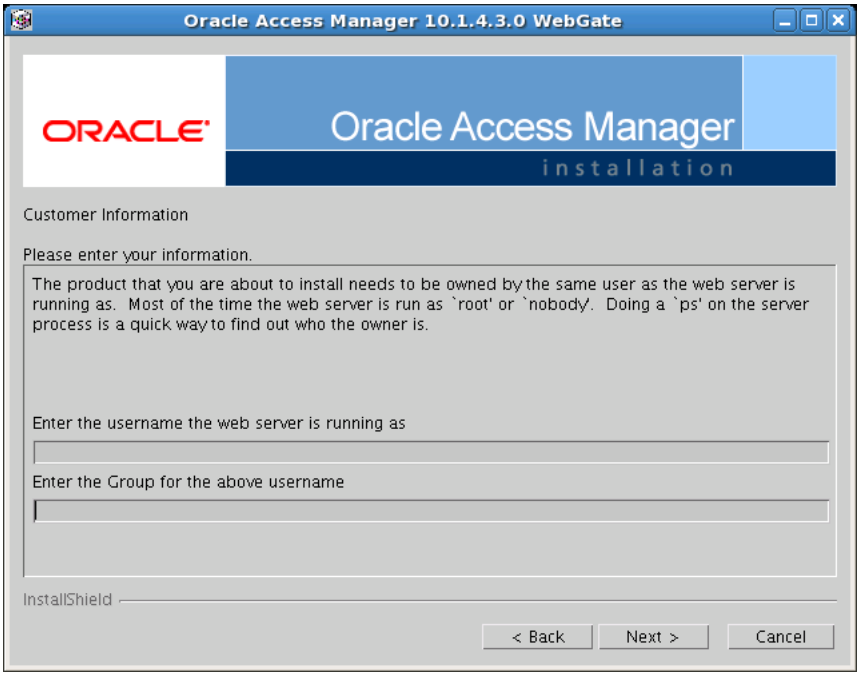
    ```
    WEBHOST2> cd ORACLE_BASE/config/CommonDomain_webtier1/bin

    WEBHOST2> ./opmnctl stopall
    WEBHOST2> ./opmnctl startall
    ```

32. From *`ORACLE_BASE`*`/repository/installers/webgate`, start the WebGate installation:

    ```
    run ./Oracle_Access_Manager10_1_4_3_0_linux64_OHS11g_WebGate -gui
    ```

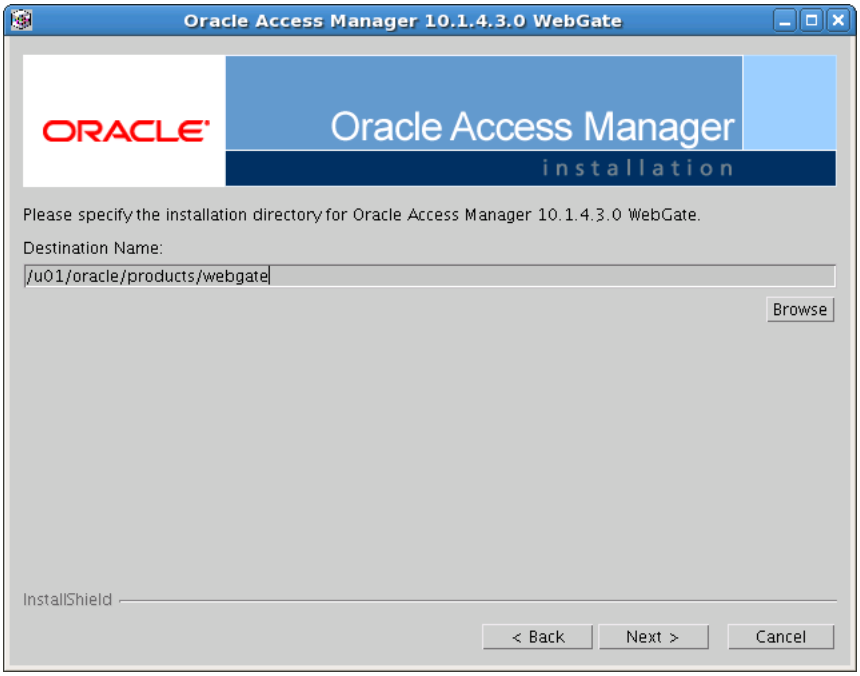
33. When the Welcome window opens, click **Next**. The Customer Information window, shown in Figure 5–22, opens.

Figure 5–22 Customer Information Window



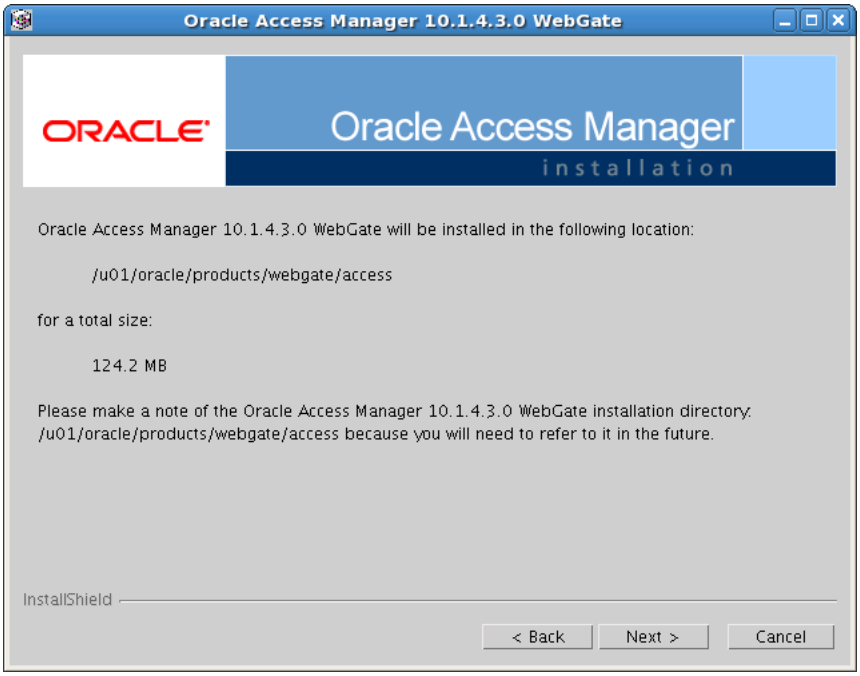
34. Enter the necessary information and click **Next**. The Installation Directory window, shown in Figure 5–23, opens.

Figure 5–23 Installation Directory Window



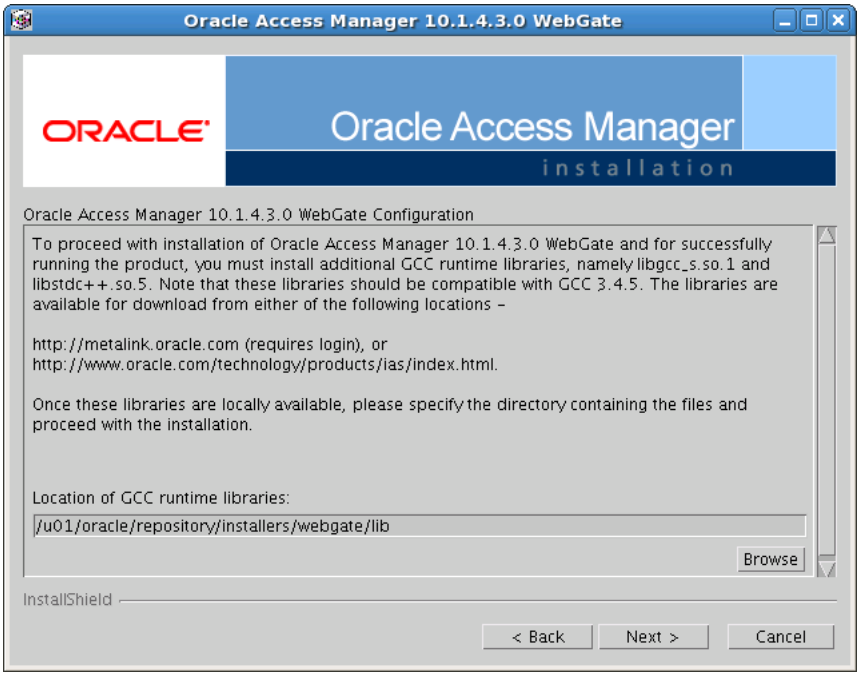
35. Enter a destination name and click **Next**. The Destination Confirmation window, shown in Figure 5–24, opens.

Figure 5–24 Destination Confirmation Window



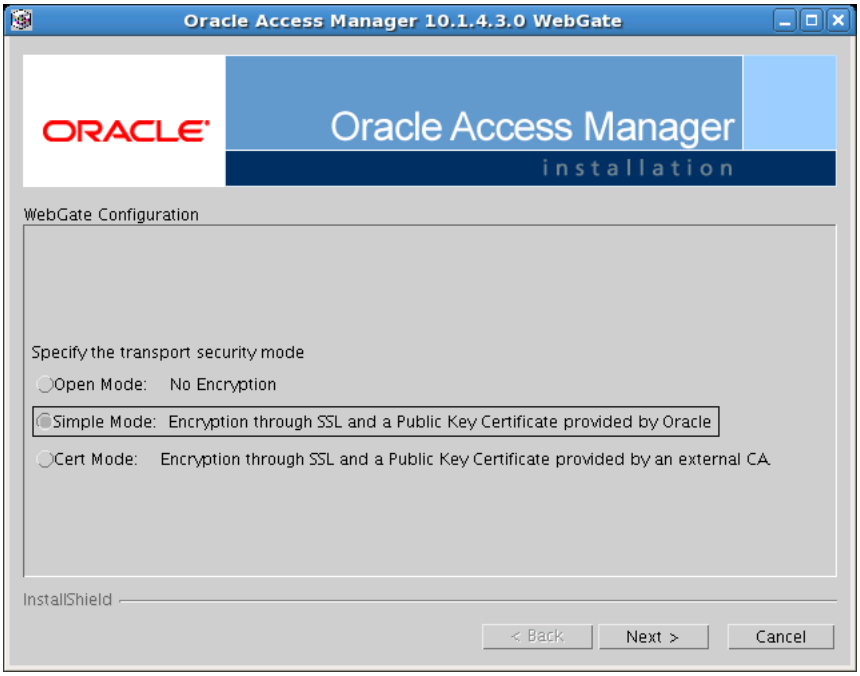
36. Click **Next** to start configuring WebGate. The window shown in Figure 5–25 opens.

Figure 5–25 WebGate Configuration (1)



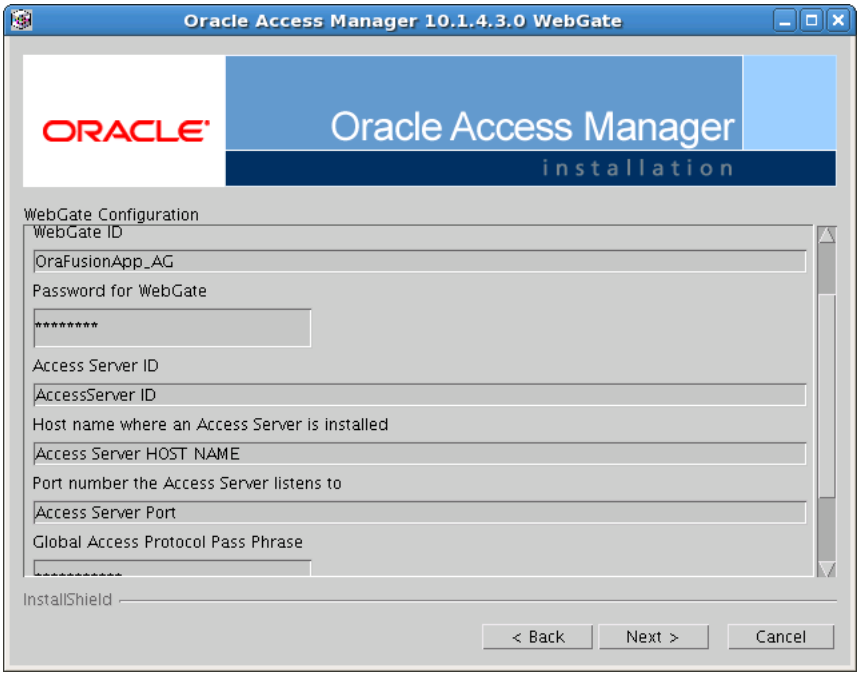
37. Enter the location for the GCC runtime libraries and click **Next**. Following an interim window, the one shown in Figure 5–26 opens.

Figure 5–26 WebGate Configuration (2)



38. Select **Simple Mode** and click **Next**. The window shown in Figure 5–27 opens.

Figure 5–27 WebGate Configuration (3)

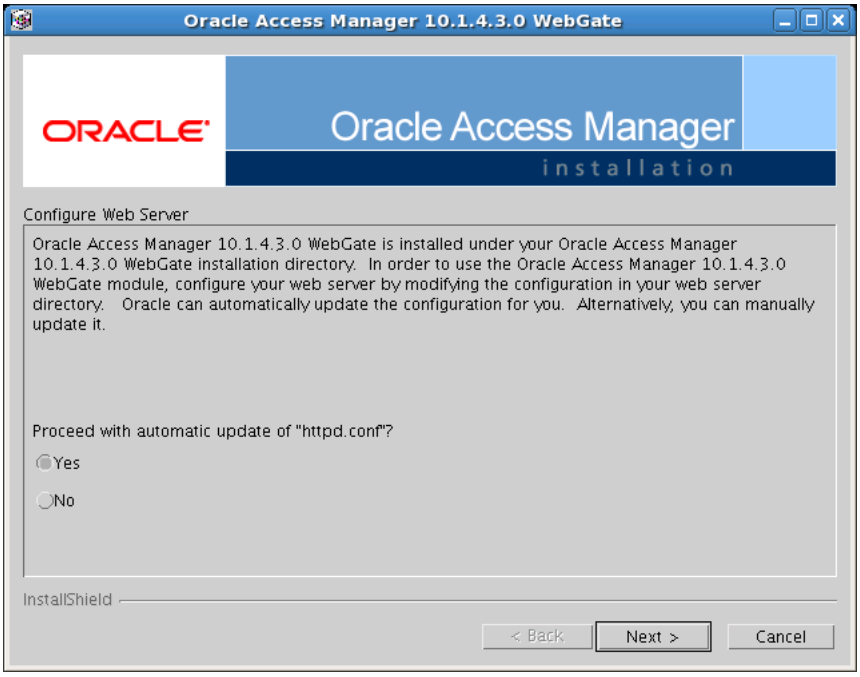


Note: **AccessServer ID**, **Host name where an Access Server is installed**, **Port number the Access Server listens to**, and **Global Access Protocol Pass Phrase** need to be obtained from the Oracle Identity Manager stack install.

Note: The password provided for the WebGate should be same as the one in the provisioning plan.

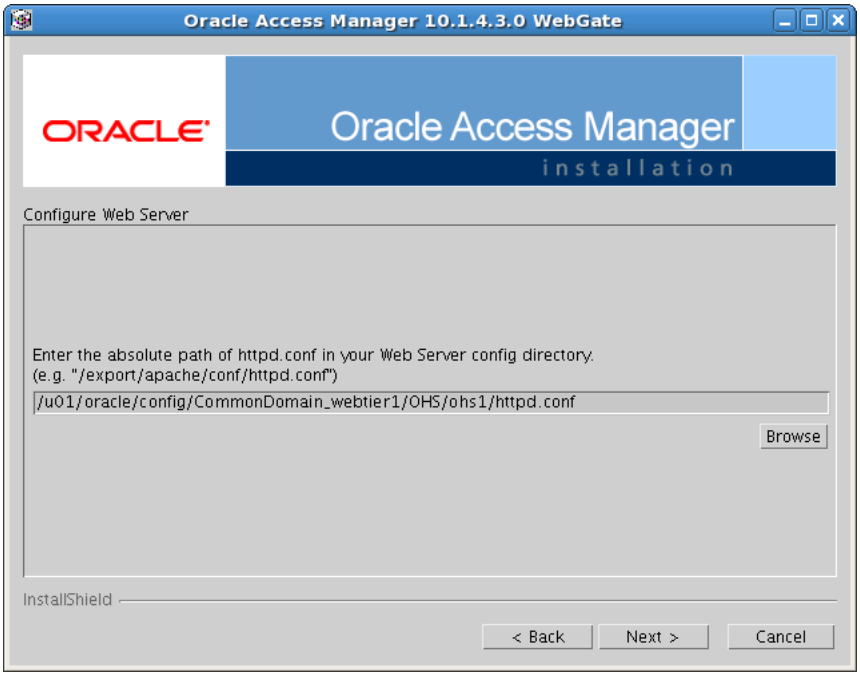
39. Enter the necessary information and click Next. Following an interim window, the one shown in Figure 5–28 opens.

Figure 5–28 WebGate Configuration (4)



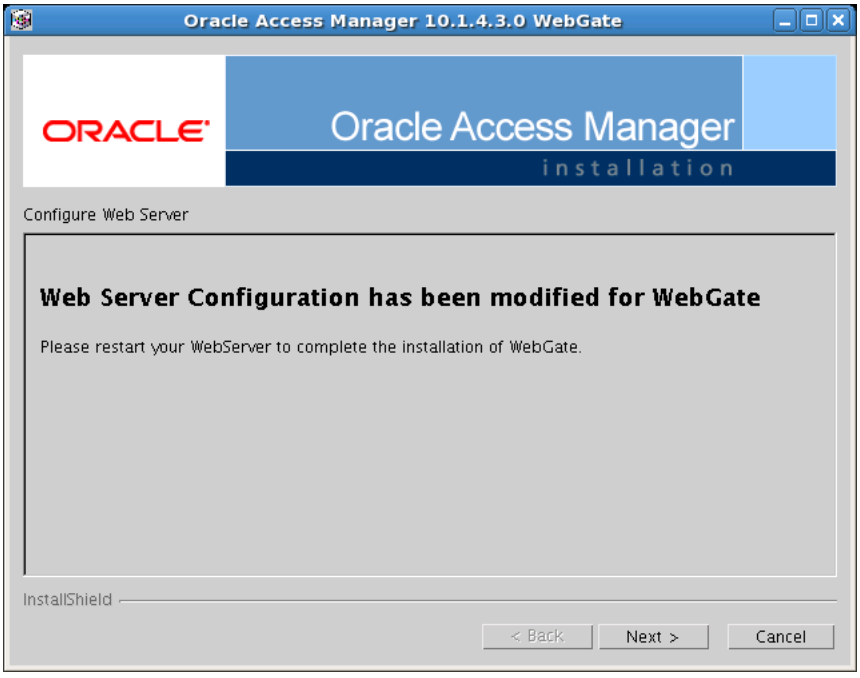
40. Select **Yes** and click **Next**. The Configure Web Server window, shown in Figure 5–29, opens.

Figure 5–29 Configure WebServer Window



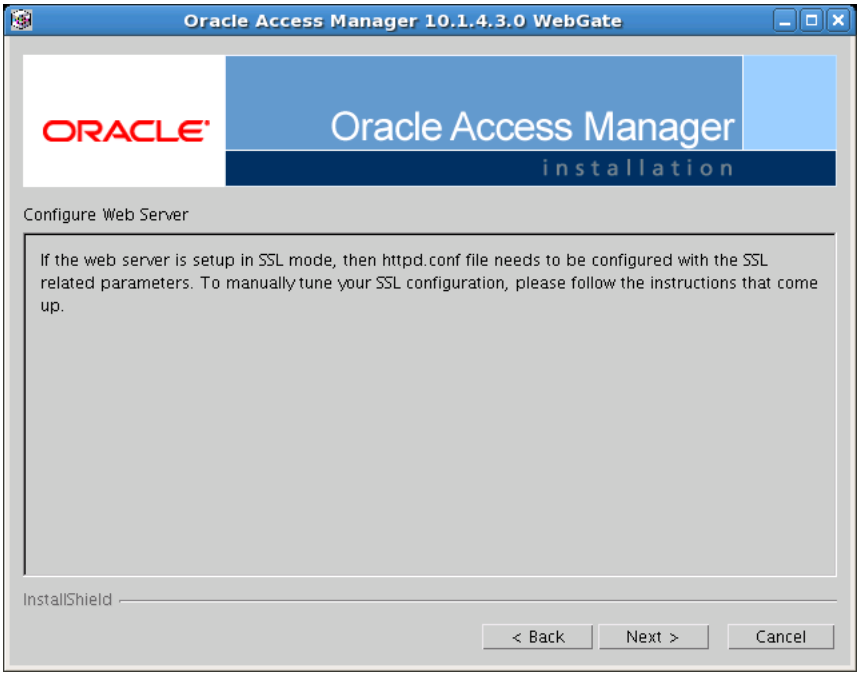
41. Browse for an absolute path or enter one, and click **Next**. A warning window, shown in Figure 5–30, opens.

Figure 5–30 WebGate Configuration Window (5)



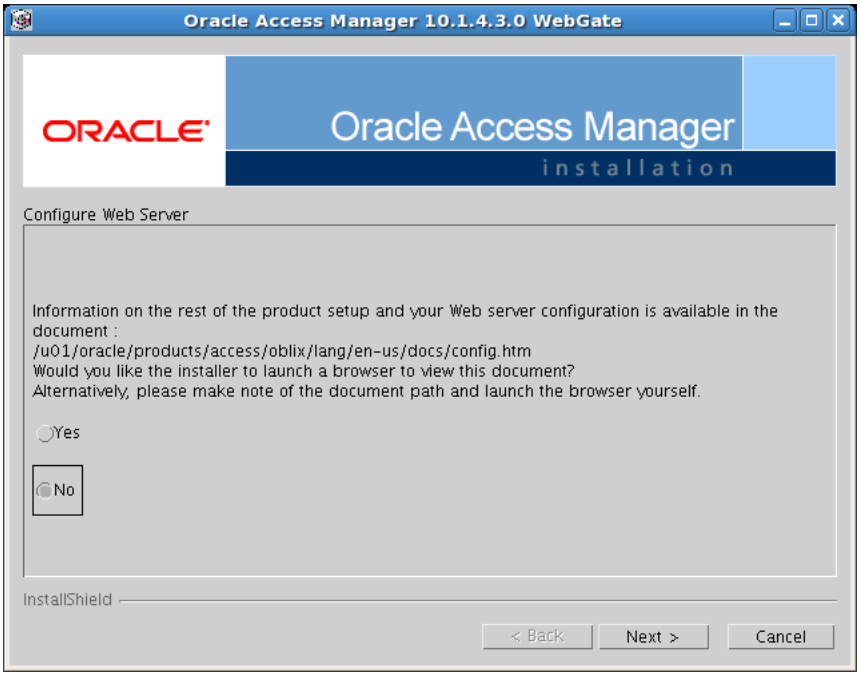
42. Click **Next** and restart the WebServer, as described in Step 31. The window shown in Figure 5–31 opens after the WebServer has been restarted.

Figure 5–31 WebServer Configuration (6)



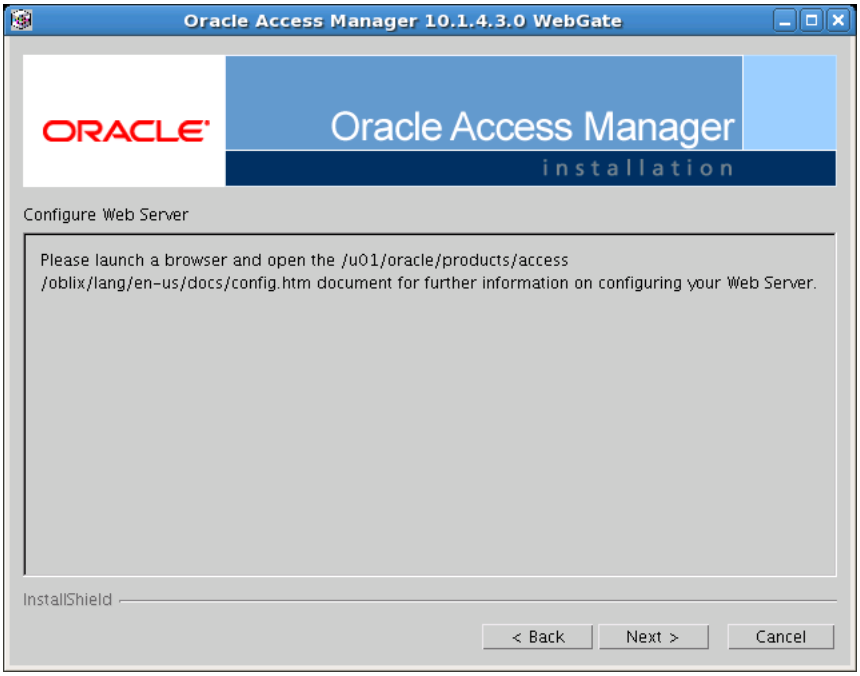
43. Click Next. The window shown in Figure 5–32 opens.

Figure 5–32 WebServer Configuration (7)



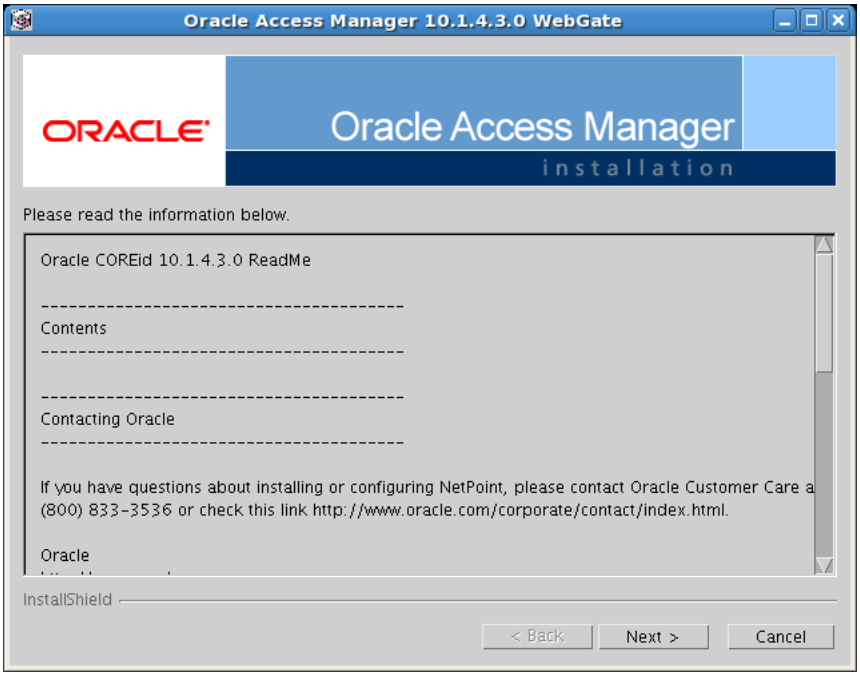
44. Select an option and click **Next**. An informational window, shown in Figure 5–33, opens.

Figure 5–33 WebServer Configuration (8)



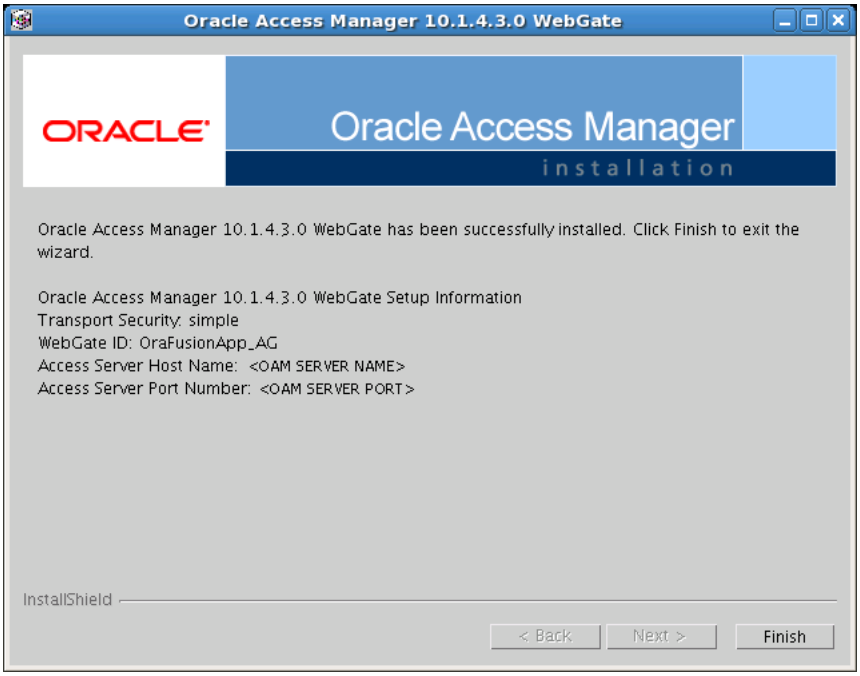
45. Click **Next**. Another informational window, shown in Figure 5–34, opens.

Figure 5–34 WebServer Configuration (9)



46. Click Next. A window indicating that the installation was successful opens. Figure 5–35 shows this window.

Figure 5–35 WebServer Configuration Window (10)



47. Click **Finish**. WebGate has now been installed.

5.2 Installing WebGate Patches

If WebGate patches need to be installed, do the following:

1. Change directory to *ORACLE_BASE*`/repository/installers/webgate/ext/Oracle_Access_Manager10_1_4_3_0_BP08_Patch_linux64_OHS11g_WebGate_binary_parameter`.
2. Run the following command, specifying the directory where `webgate/access` is installed:

   ```
   ./patchinst
   ```

3. Install all the patches in the *ORACLE_BASE*`/repository/installers/webgate/ext` directory.

After you install the patches, perform Step 31 to restart the WebServer. Oracle HTTP Server scaleout is complete. *WEBHOST1* and *WEBHOST2* should now behave identically.

5.3 Wiring Oracle HTTP Server with Load Balancer

Configure your load balancer to route all HTTP requests to the hosts running Oracle HTTP Server, that is, *WEBHOST1* and *WEBHOST2*.

You do not need to enable sticky session (insert cookie) on the load balancer when Oracle HTTP Server is the front end to Oracle WebLogic Server. You need sticky session if you are going directly from the load balancer to Oracle WebLogic Server, which is not the case in the topology described in this guide.

You also must set Monitors for HTTP.

5.4 Validating Oracle HTTP Server on WEBHOST2

To validate once the installation is complete:

1. Check that it is possible to access the Oracle HTTP Server home page using the following URLs:
   - `http://webhost1.mycompany.com:10601`
   - `http://webhost2.mycompany.com:7777`
2. Stop *WEBHOST1*:

```
WEBHOST1> cd /u01/oracle/config/CommonDomain_webtier/bin

WEBHOST1> ./opmnctl stopall
```

3. Access the following URLs to ensure that the Administration console is visible:
   - `http://crminternal.mycompany.com:7777/console`
   - `http://hcminternal.mycompany.com:7777/console`
   - `http://scminternal.mycompany.com:7777/console`
   - `http://commoninternal.mycompany.com:7777/console`
   - `http://biinternal.mycompany.com:7777/console`
4. Access the following URLs to ensure that the Oracle Fusion Applications login screen is visible:
   - `https://crmexternal.mycompany.com/sales/faces/mooOpportunityHome`
   - `https://crmexternal.mycompany.com/crmPerformance/faces/TerritoriesMain`
   - `https://crmexternal.mycompany.com/contractManagement/faces/ContractsDashboard`
   - `https://crmexternal.mycompany.com/customer/faces/CustomerCtrWorkarea`

6 Configuring Node Manager

This chapter describes how to configure Node Manager in accordance with enterprise deployment recommendations.

This chapter includes the following topics:

- Section 6.1, "Configuring Node Manager for CRMHOST2"
- Section 6.2, "Creating the Identity Keystore on CRMHOST2"

6.1 Configuring Node Manager for CRMHOST2

Do the following:

1. Run the following command:

```
CRMHOST2> cd ORACLE_BASE/products/fusionapps/wlserver_10.3/common/nodemanager
```

2. In the `/nodemanager` directory, copy the content of the node-specific directory to *CRMHOST2*. In this case, *CRMHOST1* is the node-specific directory.

```
CRMHOST2> cp -r CRMHOST1 CRMHOST2
```

3. Change directory to *CRMHOST2*. You should see the following files:

```
nm_data.properties   nodemanager.log   startNodeManagerWrapper.sh
nodemanager.domains  nodemanager.properties
```

> **Note:** Manually delete any lock files that may be present. For example, `nodemanager.log.lck`.

4. In the `nodemanager.domains` file, edit all the domain paths that are local to *CRMHOST2*. For example, `CRMDomain=/u02/local/oracle/domains/CRMHOST2/CRMDomain.`

> **Note:** Because BIDomain is a bit different, an example path would be `BIDomain=/u02/local/oracle/config/domains/CRMHOST1/BIDomain`.

5. In the `startNodeManagerWrapper.sh` file, change `NM_HOME` to *ORACLE_BASE*`/products/fusionapps/wlserver_10.3/common/nodemanager/`*CRMHOST2*.

6. In the nodemanager.properties file:
   - Ensure that the path to the local machine /u02/local/oracle/nodemanager/ exists, and that the LogFile value is pointing to /u02/local/oracle/nodemanager/*CRMHOST2*.log.
   - Ensure that the path for DomainsFile and NodeManagerHome are correct for *CRMHOST2*.
   - Add or modify the following lines:

```
KeyStores=CustomIdentityAndCustomTrust
CustomIdentityKeyStoreFileName=ORACLE_BASE/products/fusionapps/wlserver_
  10.3/server/lib/CRMHOST2_fusion_identity.jks
CustomIdentityPrivateKeyPassPhrase=keypassword
CustomIdentityAlias=CRMHOST2_fusion
```

Note: *keypassword* is the password given in the provisioning plan. (Check provisioning.setup.common.core.key.password in the provisioning plan.)

6.2 Creating the Identity Keystore on CRMHOST2

Provisioning has created the identity keystore *CRMHOST1*_fusion_identity.jks for *CRMHOST1*. Subsequently, the identity keystore *CRMHOST2*_fusion_identity.jks must be created for *CRMHOST2*.

Do the following to create the keystore:

1. Change directory to ORACLE_BASE/products/fusionapps/wlserver_10.3/server/lib.

   Ensure the *CRMHOST1*_fusion_identity.jks and fusion_trust.jks files are present.

2. Back up fusion_trust.jks to fusion_trust.jks.org.

3. Run the following command to set the CLASSPATH:

```
CRMHOST2> ORACLE_BASE/products/fusionapps/wlserver_10.3/server/bin/setWLSEnv.sh
```

   Ensure that the CLASSPATH has been set:

```
CRMHOST2> which keytool
```

   The output should point to the *ORACLE_BASE*/products/fusionapps/jdk6/jre/bin/keytool.

4. Run the following command to create the keypair for *CRMHOST2*_fusion_identity.jks:

```
CRMHOST2> keytool -genkeypair -keyalg RSA -alias CRMHOST2_fusion -keypass
keypassword -keystore CRMHOST2_fusion_identity.jks -storepass keystorepassword
-validity 180 -dname 'CN=CRMHOST2, OU=defaultOrganizationUnit,
O=defaultOrganization, C=US'
```

   where

   - *keystorepassword* is the password given in the provisioning plan (check provisioning.setup.common.core.keystore.password in the provisioning plan)

- *keypassword* is the password given in the provisioning plan (check `provisioning.setup.common.core.key.password` in the provisioning plan)

Notes:

- It is recommended to keep the commands in a file and then execute it.
- Since the passwords in the plan are encrypted, take note of or save the passwords when you are creating the plan.

---

5. Run the following command to export the certs:

```
CRMHOST2> keytool -exportcert -alias CRMHOST2_fusion
-keystore CRMHOST2_fusion_identity.jks
-storepass keystorepassword -rfc -file /tmp/appIdentityKeyStore.jks
```

Note: If the alias *CRMHOST2*`_fusion` exists, run this command to delete it:

```
keytool -delete -alias CRMHOST2_fusion -keystore fusion_trust.jks
-storepass keystorepassword
```

The following command will display the certificates in the trust keystore:

```
keytool -list -keystore fusion_trust.jks -storepass
keystorepassword
```

---

6. Run the following command to import the certs:

```
CRMHOST2> keytool -importcert -noprompt -alias CRMHOST2_fusion -file
/tmp/appIdentityKeyStore.jks -keystore fusion_trust.jks -storepass
keystorepassword
```

7. Verify that the file *CRMHOST2*`_fusion_identity.jks` has been created in the directory *ORACLE_BASE*`/products/fusionapps/wlserver_10.3/server/lib` directory.
8. Start Node Manager on *CRMHOST2* by running the following command:

```
ORACLE_BASE/products/fusionapps/wlserver_10.3/common/nodemanager/CRMHOST2/
startNodeManagerWrapper.sh &
```

7

Scaling Out the Oracle Fusion Customer Relationship Management Domain

This chapter describes how to scale out the Oracle Fusion Customer Relationship Management domain.

This chapter includes the following topics:

- Section 7.1, "Overview of the Oracle Fusion Customer Relationship Management Domain"
- Section 7.2, "Prerequisites for Scaling Out the Oracle Fusion Customer Relationship Management Domain"
- Section 7.3, "Adding a New Machine in the Oracle WebLogic Server Console"
- Section 7.4, "Packing and Unpacking the Managed Server Domain Home"
- Section 7.5, "Cloning Managed Servers and Assigning Them to CRMHOST2"
- Section 7.6, "Configuring Data Quality for Scale Out"
- Section 7.7, "Oracle HTTP Server Configuration"
- Section 7.8, "Validating the System"

7.1 Overview of the Oracle Fusion Customer Relationship Management Domain

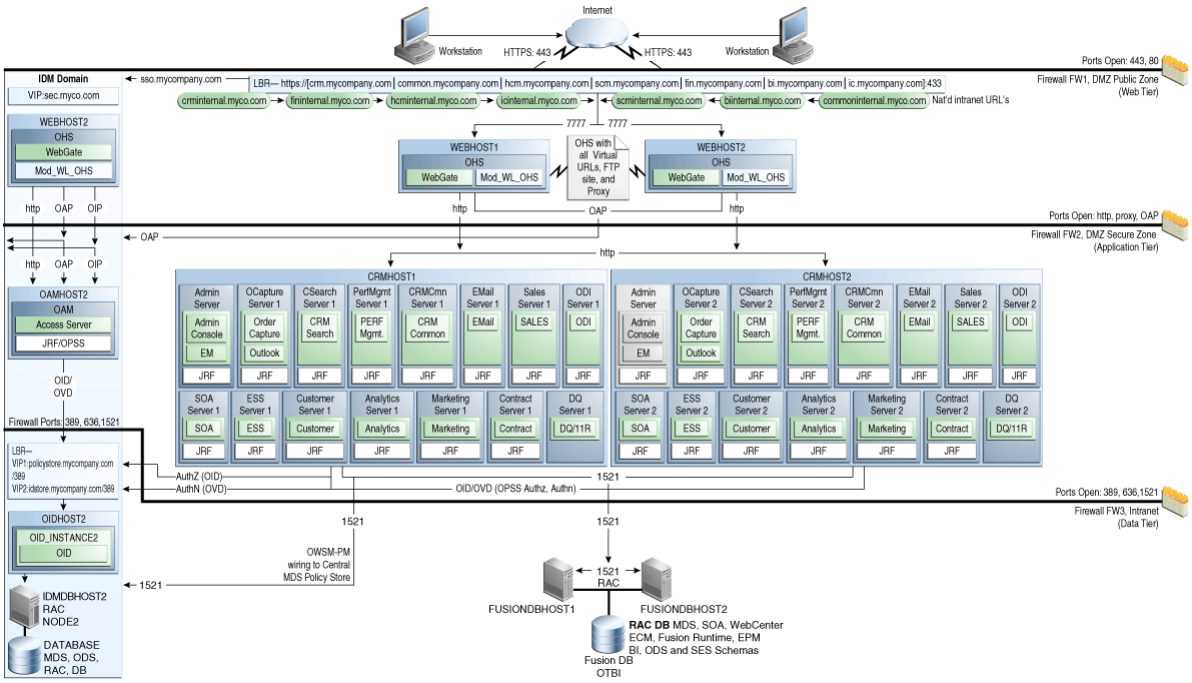
The Oracle Fusion Customer Relationship Management application is a very distributed and modularized one. Applications within Oracle Fusion Customer Relationship Management, which are deployed on the domain, are the following:

- Marketing
- Sales
- Territory Management
- Contract Management
- Customer Management
- Order Capture
- Email Marketing
- OutLook/Mobile
- CRM Common

In addition to the applications, the Oracle Fusion Customer Relationship Management domain also contains Oracle Fusion Customer Relationship Management Analytics, which is the Oracle BI Enterprise Edition broker application that interfaces with Oracle Application Development Framework, Oracle BI Enterprise Edition, and Oracle Data Integrator agent for data import flow.

Figure 7–1 shows the Oracle Fusion Customer Relationship Management domain within the overall reference enterprise deployment topology.

Figure 7–1 Reference Topology for Oracle Fusion Customer Relationship Management Domain



7.2 Prerequisites for Scaling Out the Oracle Fusion Customer Relationship Management Domain

Before you begin, ensure the following:

- Node Manager has been started in the Secure Sockets Layer (SSL) mode by following the instructions in Chapter 6, "Configuring Node Manager"
- You are starting with a clean machine if it is the first time it is being used for a scale out
- The `/etc/hosts` file has proper entries. To verify, ping this machine with the fully qualified name of the machine
- The user created on *CRMHOST2* should the same as the user on *CRMHOST1*
- The directory structure `/u01/oracle` is mounted to same shared file system as *CRMHOST1*
- The directory structure `/u02/local/oracle/config` on *CRMHOST2* has been created
- The initial Oracle Fusion Customer Relationship Management deployment on *CRMHOST1* has already been done and verified by provisioning

7.3 Adding a New Machine in the Oracle WebLogic Server Console

To add a new machine:

1. Log in to the Administration Server: `http://crminternal.mycompany.com:7777/console`.
2. Navigate to **CRMDomain > Environment > Machines**.

   LocalMachine is located in the right-hand pane.
3. In the left-hand pane, click **Lock & Edit**.
4. In the right-hand pane, first click **New** to add the remote machine, and then specify the following:
   - Name - enter *CRMHOST2*
   - Machine operating system - Unix
5. Click **Next**.
6. In the window that opens, set the following attributes:
   - Type - SSL
   - Listen Address - <*CRMHOST2*>

     Note: The "localhost" default value here is wrong.
   - Listen port - 5556
7. Click **Finish** and activate the changes.

Note: If you get an error when activating the changes, see Section 18.8.18, "Administration Console Redirects from Internal URL to Container URL after Activation" for the temporary solution.

7.4 Packing and Unpacking the Managed Server Domain Home

Since the *CRMHOST1* domain directory file system is also available from *CRMHOST2*, both the `pack` and `unpack` commands can be executed from the *CRMHOST2*.

1. Do the following:
   a. Change directory to *ORACLE_BASE*`/products/fusionapps/oracle_common/common/bin`.
   b. Run the `pack` command:

      ```
      CRMHOST2> ./pack.sh -managed=true -domain=ORACLE_BASE/config/domains/
      CRMHOST1/CRMDomain -template=ORACLE_BASE/user_templates/
      CRMDomain_managed.jar -template_name="CRM_Managed_Server_Domain"
      ```
2. Ensure that `/u02/local/oracle/config/domains/`*CRMHOST2*`/CRMDomain` is empty, and then run the `unpack` command:

```
CRMHOST2> ./unpack.sh -domain=/u02/local/oracle/config/domains/
CRMHOST2/CRMDomain -template=ORACLE_BASE/user_templates/CRMDomain_managed.jar
```

Here, *ORACLE_BASE* is shared, and `/u02/local` is local to *CRMHOST2*.

7.5 Cloning Managed Servers and Assigning Them to CRMHOST2

To add a managed server and assign it to *CRMHOST2*:

1. Log in to the Administration Server: `http://crminternal.mycompany.com:7777/console`.
2. Navigate to **CRMDomain > Environment > Servers**.
3. Switch to **Lock & Edit** mode.
4. Select the ***Managed_Servers*** checkbox (for example, **SalesServer_1**) and then click **Clone**.
5. Specify the following server identity attributes:
   - Server Name - `SalesServer_2`

     Note: To ensure consistency in naming, copy the name of the server shown in **Server Identity** and paste it into the **Server Name** field. Then change the number to "_2".

   - Server Listen Address - <*CRMHOST2*>
   - Server Listen Port - leave "as is"
6. Click **OK**.

   You now should see the newly cloned sales server, `SalesServer_2`.
7. Click **SalesServer_2** and change the following attributes:
   - Machine - <*CRMHOST2*>
   - Cluster Name - Default, SalesCluster
8. Click **Save** and then **Activate Changes**.
9. From the **Name** column, click the **SalesServer_2** scaled-out server link.
10. Click **Lock & Edit**, and then choose the **Keystores** tab.
11. Ensure that the keystores value is **Custom Identity and Custom Trust**.
12. Do the following:
    a. Change the Custom Identity Keystore path to point to the *ORACLE_BASE*/`products/fusionapps/wlserver_10.3/server/lib/`*CRMHOST2*`_fusion_identity.jks` file.
    b. Leave the Custom Identity Keystore type blank.
    c. Change the Custom Identity Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."
    d. Re-enter the Confirm Custom Identity Keystore Passphrase.

e. Ensure that the Confirm Custom Trust Keystore path is pointing to the *ORACLE_BASE*/products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks file.

f. Leave the Custom Trust Keystore type blank.

g. Change the Custom Trust Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

h. Re-enter the Custom Trust Keystore Passphrase.

i. Click **Save**.

13. Choose the **SSL** tab.

a. Make sure that Identity and Trust Locations is set to **Keystores**.

b. Change the Private Key Alias to *CRMHOST2*_fusion.

c. Change the Private Key Passphrase to the *keypassword*, as described in the second bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

d. Re-enter the *keypassword* from Step c for the Confirm Private Key Passphrase.

e. Click **Save**.

14. Click **Activate Changes**.

15. Repeat Steps 2 to 14 for all the managed servers on this domain.

16. Set the following environment variable:

```
WLST_PROPERTIES="-Dweblogic.security.SSL.trustedCAKeyStore=ORACLE_BASE/
products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks"
```

17. Stop the domain's Administration Server:

```
CRMHOST1> ORACLE_BASE/config/domains/CRMHOST1/CRMDomain/bin/stopWebLogic.sh
```

18. Restart the domain's Administration Server:

```
CRMHOST2> ORACLE_BASE/products/fusionapps/wlserver_10.3/common/bin/wlst.sh

CRMHOST2> nmConnect(username='username', password='password',
domainName='CRMDomain', host='CRMHOST1',port='5556',
nmType='ssl', domainDir='ORACLE_BASE/config/domains/CRMHOST1/CRMDomain')

CRMHOST2> nmStart('AdminServer')
```

Note: The *username* and *password* used in the `nmConnect` are the Node Manager credentials (username and password) specified when creating the provisioning plan. This is shown in Figure 4–3 in "Using the Provisioning Process to Install Components for an Enterprise Deployment".

19. Run the newly created managed servers:

a. Navigate to **CRMDomain > Environment > Servers > Control**.

b. Select the newly created managed servers and click **Start**.

c. Navigate to **CRMDomain > Environment > Servers** and check the **State** to verify that the newly created managed servers are running.

20. Log in to the Administration Server once again (`http://crminternal.mycompany.com:7777/console`) and verify that all the managed servers, including scaled-out servers, are running.

Note: For all the scaled-up and scaled-out servers, change the Arguments in the `/u02/local/oracle/config/domains/HOSTNAME/DomainName/servers/ManagedServer/data/nodemanager/startup.properties` file to the following:

```
Arguments=-DJDBCProgramName\=DS/CRMDomain/SalesServer_2
-Dserver.group\=SalesCluster
```

Note: For all the scaled-up and scaled-out managed servers, do the following:

1. Access the Oracle WebLogic Server Administration Console for the CRMDomain:

   `http://crminternal.mycompany.com:7777/console`

2. Navigate to **Environment > Servers** and click the "Managed Server" link.
3. First select the **Logging** tab and then the **HTTP** tab.
4. Update the following parameters:

   – Log file name: logs/access.log.%yyyyMMdd%

   – Rotation Type: By Time

   – Limit number of retained files: leave this option unchecked

   – Rotate log file on startup: leave this option unchecked

5. Click **Save**.
6. Expand **Advanced Node** and set the following:

   – Format: Extended

   – Extended Logging Format Fields:

   ```
   date time time-taken cs-method cs-uri sc-status
   sc(X-ORACLE-DMS-ECID) cs(ECID-Context) cs(Proxy-Remote-User)
   cs(Proxy-Client-IP)
   ```

7. Click **Save** and **Activate Changes**.
8. Restart the Managed Server for the changes to take affect.

7.6 Configuring Data Quality for Scale Out

Data Quality is an Informatica tool that provides the following:

- A *matching* function, which allows you to search, match, and identify duplicate customer records based on key customer attributes such as name, address, date of birth, Social Security Number, or tax ID number. It also includes data quality connector, a generalized interface that is capable of making real-time and batch requests to an external data-cleansing engine.
- An *address-cleansing* feature, which corrects and validates address data based on postal requirements.

Implementing Data Quality requires four steps:

1. Obtaining postal reference data and license keys.
2. Setting up the Data Quality Engine, also known as Informatica Identity Resolution (IIR).
3. Configuring Data Quality Connector and IIR in Oracle Fusion Functional Setup Manager.
4. Creating a second Data Quality server on *CRMHOST2*.

7.6.1 Obtaining Postal Reference Data and License Keys

Prior to configuring data quality for scale out, you must obtain postal reference data and license key files from AddressDoctor. For information about how to obtain these, see the licensing documentation.

7.6.2 Setting Up the Data Quality Engine

Do the following:

1. On *CRMHOST1*, change directory to *ORACLE_BASE*/products/InformaticaIR/bin.
2. Run the following commands:

```
CRMHOST1> bash
CRMHOST1> . ./setfusionEnv.sh
CRMHOST1> ./liup
CRMHOST1> ./idsup
```

3. Start `./idsconc -a`.
4. On the launched console, select **Rulebase Type=SSA** and **Alias=rtunitrb** and click **OK**. Then click **Yes** to create a new rulebase.
5. In the IIR console, navigate to **System > New > Create a system from SDF**.
6. Enter the following system information:
   - **System Name** - `FusionDQRealtime`
   - **System Definition File** - *ORACLE_BASE*`/products/InformaticaIR/ids/FusionDQRealtime.sdf`
   - **DB Type** - `SSA`
   - **Alias** - `fusion_s01`
7. Navigate to **System menu > Select > System Name: FusionDQRealtime** and click **OK**.
8. Navigate to **System menu > Select > Load IDT** to start loading the individual IDT tables, one by one, until the following IDTs are completed:

```
load_location
load_organization
load_org_address
load_person
load_per_address
load_per_phone
```

9. Close the IIR Console started in Step 2 and run the following commands to stop the IIR server:

```
ORACLE_BASE/products/InformaticaIR/bin/idsdown
```

```
ORACLE_BASE/products/InformaticaIR/bin/lidown
```

10. Assuming you have received the postal reference data and license key files from AddressDoctor (that you requested in Section 7.6.1), run the following commands:

```
ORACLE_BASE/productsInformaticaIR/bin > cd ORACLE_BASE/products/InformaticaIR/
ssaas/ad5/ad/db

ORACLE_BASE/products/InformaticaIR/ssaas/ad5/ad/db > cp mylocation/key .

ORACLE_BASE/products/InformaticaIR/ssaas/ad5/ad/db > cp mylocation/*.MD .
```

    mylocation is the where you copied the license key and the postal reference data.

    `*.MD` is the postal data reference file. `key` is a text file that contains the AddressDoctor license.

    AddressDoctor supports 248 countries. Each `*.MD` file is per country or a group of countries. Each of these files should be copied to the `*.MD` directory.

11. Run the following commands to start the IIR server:

```
ORACLE_BASE/products/InformaticaIR/bin/liup
ORACLE_BASE/products/InformaticaIR/bin/idsup
```

12. From the IIR Console, do the following to start the UPD Synchronizer:

    a. Run the following command:

```
ORACLE_BASE/products/InformaticaIR/bin/idsconc -a
```

    b. On the launched console, select **Rulebase Type=SSA** and **Alias=rtunitrb**, and click **OK** to go the RuleBase.

    c. In the IIR console, go to **System > Choose Existing System** select FusionDQRealtime.

    d. Go to **Tools > UPD Synchronizer > Run Synchronizer** and click **OK** to accept all the defaults shown in the Update Synchronizer popup window.

Note: Ensure that the **IDT Name=(all)** option is selected.

7.6.3 Configuring the Data Quality Connector and IIR

Do the following:

1. Log in to Oracle Fusion Functional Setup Manager as an administrator. For example, `http://commoninternal.mycompany.com:7777/homePage/faces/AtkHomePageWelcome`.
2. Search for the task "Manage Data Quality Server Configuration."
3. Click **Go to Task** to open the Manage Data Quality Server Configurations page, and then click **Search** to find all existing configurations.
4. Select **Realtime and Batch Basic Server** and click **Edit**.
5. Enter the server IP address and port of the IIR search server you set up as the IIR matching server.
6. Select **Realtime Cleanse Server** , click **Edit**, and repeat Step 5.
7. Select **Batch Cleanse Server**, click **Edit**, and repeat Step 5.

8. Search for the task "Manage Data Quality Synchronization Configuration".
9. Click **Refresh Identity Table Information** to refresh IDT repository information.
10. Select **Enable for Sync** for each IDT.
11. Click **Schedule Synchronization Process** and then **Advanced**.
12. Select **Using a schedule**.
13. Select **Hourly/Minute** from the **Frequency** dropdown list. (This frequency should be determined by the business requirement.)
14. Enter "every 5 minutes" for this example. (This "every 5 minutes" sample should be determined by the business requirement.)
15. Choose a "next few days" end date from the calender. (This "few days" sample should be determined by the business requirement.)
16. Submit the scheduled Sync ESS job.
17. Do the following to validate that Data Quality is up and running:
    a. Ensure that the Customer Center application is running.
    b. In Customer Center, create a unique organization and then try to create that same organization again.

       If Data Quality is woking correctly, a popup window will display telling you that you cannot create that organization because it already exists.

7.6.4 Creating a Second Data Quality Server on CRMHOST2

Do the following:

1. Run the following command:

```
CRMHOST2> cd ORACLE_BASE/respository/installers/iir/fusion_iir
```

2. Edit the `install.props` file to include the following values:

```
##############################################
# USE ABSOLUTE PATHS FOR ALL
##############################################       webtier_patchset
# These environment variables are required to be set for
# IIR to be able to use the ORACLE DB CLIENT bgate
##################################################logic
ORACLE_HOME=ORACLE_BASE/products/dbclient
TNS_ADMIN=ORACLE_BASEproducts/dbclient/network/admin
LD_LIBRARY_PATH=ORACLE_BASE/products/dbclient/lib
##################################################Installer.bat  unInstaller.bat
# These properties are required to be set fornInstaller.sh   unInstaller.sh
# IIR to be installed in the right directory st_console
##############################################
FUSION_STAGE_DIR=ORACLE_BASE/repository/installers/iir
IIR_STAGE_DIR=ORACLE_BASE/oracle/repository/installers/iir
IIR_VERSION=IIR_901sp1_linux_amd64
IIR_TOP=/ORACLE_BASE/products/IIR2
##############################################
# These properties are required to be set for IIR to be
# able to connect to the Oracle DB to install Schema Objects
##############################################
IIR_DB_HOSTNAME=DB HOST NAME
IIR_DB_PORT=1521
IIR_DB_SID=fusiondb1
```

```
IIR_DB_USER=fusion_dq
IIR_DB_PASSWD=PASSWORD
################################################
# ALL THESE PROPERTIES ARE NEEDED BY THE INSTALLER
# DO NOT MODIFY THESE UNLESS NECESSARY
################################################
IIR_INSTALL_LOG_LOC=ORACLE_BASE/repository/installers/iir/fusion_iir
IIR_INSTALL_TYPE=all
IIR_INCLUDE_DOC=yes
###############################################
# Default is one Server Instance
###############################################
IIR_INSTANCE_1_PORT=1660
###############################################
# Not Implemented yet
###############################################
IIR_INSTANCE_2_PORT=
###############################################
# This option is needed if you want Search to be accessible through a Browser
###############################################
IIR_HTTP=Y
###############################################
# This is for Installing the IIR Control and Sync Objects ( needed only for the
#first time)
###############################################
INSTALL_IIR_OBJECTS=N
###############################################
# This is needed for OEM Key
###############################################
SSALI_MZXPQRS=STANISLAUS
```

3. Run the following command:

   ```
   ./runInstaller.sh install.props > test_console
   ```

   The installation is now complete.

4. Run the `create_secondary_server.sql` script in *ORACLE_BASE*`/repository/fusion_iir/iir/sql`, passing the following parameters when prompted:

   - **Host** - *CRMHOST2*
   - **Port** - `IIR Secondary Port`
   - **Server Operation Code** - `IIR_RT_AND_BT_BASIC_MATCH`
   - **Server Number** - `1`
   - **Create Server Only** - `N` (this creates all required Secondary IIR Server related parameters)

   If Cleansing needs to be load balanced, run the `create_secondary_server.sql` script twice using the same parameters detailed in this step each time. However, run the **Server Operation Code** parameter with the following values:

   - **Server Operation Code** - `IIR_RT_CLEANSE`
   - **Server Operation Code** - `IR_BT_CLEANSE`

   Also make sure that all the Postal Directories and the License code are set up on *CRMHOST2*, as per the Cleansing Setup.

5. Start IIR on *CRMHOST1*, first by starting the Rulebase server, and then by starting the Search server:

```
ORACLE_BASE/products/InformaticaIR/bin > bash

ORACLE_BASE/products/InformaticaIR/bin > . ./setfusionEnv.sh

ORACLE_BASE/products/InformaticaIR/bin > . ../env/isss

ORACLE_BASE/products/InformaticaIR/bin > ./ssasrsv –m{RBPORT1}
-gFusionRBSG,ssa:rtunitrb -w1 -1$SSATOP/iirlog/priRB.log -2$SSATOP/iirlog/
priRB.err -3$SSATOP/iirlog/priRB.dbg

ORACLE_BASE/products/InformaticaIR/bin >./ssasrsv -n{SEPORT1}
-gFusionRBSG,ssa:rtunitrb -1$SSATOP/iirlog/priSE.log -2$SSATOP/iirlog/priSE.err
-3$SSATOP/iirlog/priSE.dbg
```

6. Start IIR on *CRMHOST2*, first by starting the Rulebase server, and then by starting the Search server:

```
ORACLE_BASE/products/InformaticaIR/bin > bash

ORACLE_BASE/products/InformaticaIR/bin > . ./setfusionEnv.sh

ORACLE_BASE/products/InformaticaIR/bin > . ../env/isss

ORACLE_BASE/products/InformaticaIR/bin >./ssasrsv –m{RBPORT2}
-gFusionRBSG,ssa:rtunitrb -w1 -1$SSATOP/iirlog/secRB.log -2$SSATOP/iirlog/
 secRB.err -3$SSATOP/iirlog/secRB.dbg

ORACLE_BASE/products/InformaticaIR/bin > ./ssasrsv -n{SEPORT2}
-gFusionRBSG,ssa:rtunitrb -1$SSATOP/iirlog/secSE.log -2$SSATOP/iirlog/secSE.err
-3$SSATOP/iirlog/secSE.dbg
```

7. Restart the Oracle Fusion Applications instances (which require DQ) so that the DQ Connector can do load balancing and failover.

Notes:

- RBPORT1 is the port for the Rulebase server on IIR Host 1; RBPORT2 is the port for IIR Host 2. The values for these parameters should be the value of SSA_RBPORT in `InformaticaIR/env/isss` on Hosts 1 and 2, respectively.
- SEPORT1 is the port for the Search server on IIR Host 1; SEPORT2 is the port for IIR Host 2. The values for these parameters should be the value of SSA_SEPORT in `InformaticaIR/env/isss` on Hosts 1 and 2, respectively.

For more information about data quality and IIR, see the following sections in the "Customer Relationship Management" chapter in the *Oracle Fusion Applications Post-Installation Guide*:

- "Setting Up Informatica Identity Resolution"
- "Informatica Identity Resolution Setup: Procedures"

7.7 Oracle HTTP Server Configuration

To configure Oracle HTTP Server:

1. On *WEBHOST1*:
   a. Change directory to *ORACLE_BASE*/config/CommonDomain_webtier/config/OHS/ohs1/moduleconf.
   b. Copy FusionVirtualHost_crm.conf to FusionVirtualHost_crm.conf.org.
2. Edit the FusionVirtualHost_crm.conf file, adding the scaled-out host and port to all the WebLogic Application Clusters. Example 7–1 shows sample code for SalesServer.

Example 7–1 Sample "SalesServer" Code

```
<Location /sales>
    SetHandler weblogic-handler
    WebLogicCluster <CRMHOST1:port>,<CRMHOST2:port>
</Location>
```

3. Repeat Step 2 for all applications.
4. Restart Oracle HTTP Server: cd to *ORACLE_BASE*/config/CommonDomain_webtier/bin and enter the following:

```
WEBHOST1> ./opmnctl stopall
WEBHOST1> ./opmnctl startall
```

5. Repeat Steps 1 through 4 on *WEBHOST2*.

7.8 Validating the System

You should verify URLs to ensure that the appropriate routing and failover is working from Oracle HTTP Server to the SalesCluster.

To verify the URLs:

1. Log in to the CRMDomain Oracle WebLogic Server Administration Console and stop all the managed servers on the *CRMHOST1* while the managed servers on *CRMHOST2* are running.
2. Access the following URLs to verify that routing and failover are functioning properly. (Ensure the log in prompt is visible.)
   - https://crmexternal.mycompany.com/sales/faces/mooOpportunityHome
   - https://crmexternal.mycompany.com/crmPerformance/faces/TerritoriesMain
   - https://crmexternal.mycompany.com/contractManagement/faces/ContractsDashboard
   - https://crmexternal.mycompany.com/customer/faces/CustomerCtrWorkarea
   - https://crmexternal.mycompany.com/marketing/faces/LeadsDashboard
   - https://crmexternal.mycompany.com/orderCapture/faces/SalesCatalogAdmin

3. Log in to the `CRMDomain` Oracle WebLogic Server Administration Console and stop all the managed servers on *`CRMHOST2`*.
4. Start the managed servers on *`CRMHOST1`*.
5. Repeat Step 2. (Ensure the log in prompt is visible.)
6. Start all the managed servers on *`CRMHOST2`* and verify that they are running on *`CRMHOST1`*.

8

Scaling Out the Oracle Fusion Common Domain

This chapter describes how to scale out the Oracle Fusion Common domain.

This chapter includes the following topics:

- Section 8.1, "Overview of the Oracle Fusion Common Domain"
- Section 8.2, "Prerequisites for Scaling Out the Oracle Fusion Common Domain"
- Section 8.3, "Adding a New Machine in the Oracle WebLogic Server Console"
- Section 8.4, "Scaling Out Oracle Universal Content Management"
- Section 8.5, "Packing and Unpacking the Managed Server Domain Home"
- Section 8.6, "Cloning Managed Servers and Assigning Them to CRMHOST2"
- Section 8.7, "Oracle HTTP Server Configuration"
- Section 8.8, "Validating the System"

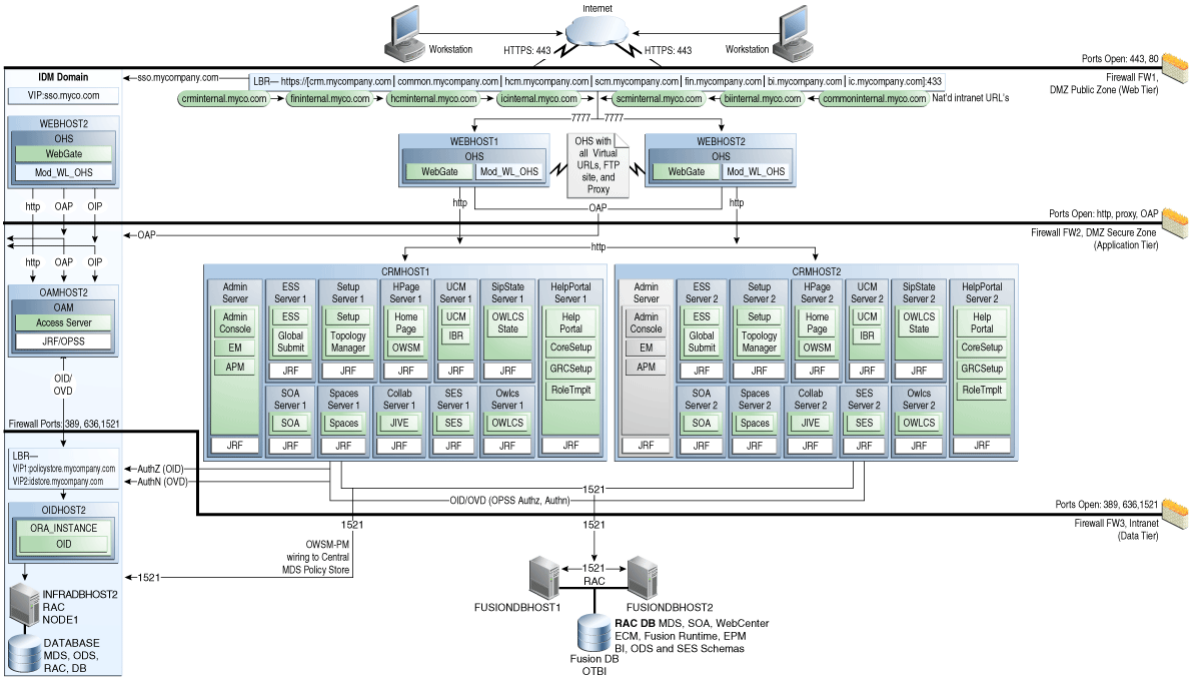
8.1 Overview of the Oracle Fusion Common Domain

The Oracle Fusion Common domain is shared by all Oracle Fusion Applications product families. Oracle Fusion Customer Relationship Management implementation is dependent on the Oracle Fusion Common domain for the following components:

- Oracle Fusion Functional Setup Manager, for all setup task flows
- Help Portal, for centralized help
- Home page application, which has the Oracle Fusion Customer Relationship Management home page and launch pad
- Oracle Universal Content Management (Oracle UCM), to store all marketing collateral as well as all attachments. Import flow also uses Oracle UCM to stage the CSV files that theuser uploads
- Oracle Secure Enterprise Search
- Oracle WebCenter Community Space and forums
- Oracle WebLogic Communication Services (OWLCS) and Oracle WebLogic SIP Server

Figure 8–1 shows the Oracle Fusion Common domain within the overall reference enterprise deployment topology for Oracle Fusion Applications.

Figure 8–1 Reference Topology for Oracle Fusion Common Domain



8.2 Prerequisites for Scaling Out the Oracle Fusion Common Domain

Before you begin, ensure the following:

- Node Manager has been started in the Secure Sockets Layer (SSL) mode by following the instructions in Chapter 6, "Configuring Node Manager"
- You are starting with a clean machine if it is the first time it is being used for a scale out
- The `/etc/hosts` file has proper entries. To verify, ping this machine with the fully qualified name of the machine
- The user created on *`CRMHOST2`* should the same as the user on *`CRMHOST1`*
- The directory structure `/u01/oracle` is mounted to same shared file system as *`CRMHOST1`*
- The directory structure `/u02/local/oracle/config` on *`CRMHOST2`* has been created
- The initial Oracle Fusion Customer Relationship Management deployment on *`CRMHOST1`* has already been done and verified by provisioning

8.3 Adding a New Machine in the Oracle WebLogic Server Console

To add a new machine:

1. Log in to the Administration Server: `http://`*`commoninternal`*`.mycompany.com:7777/console`.
2. Navigate to **CommonDomain > Environment > Machines**.

   LocalMachine is located in the right-hand pane.
3. In the left-hand pane, click **Lock & Edit**.

4. In the right-hand pane, first click **New** to add the remote machine, and then specify the following:
   - Name - enter *CRMHOST2*
   - Machine operating system - Unix
5. Click **Next**.
6. In the window that opens, set the following attributes:
   - Type - SSL
   - Listen Address - *<CRMHOST2>*

     Note: The "localhost" default value here is wrong.

   - Listen port - 5556
7. Click **Finish** and activate the changes.

Note: If you get an error when activating the changes, see Section 18.8.18, "Administration Console Redirects from Internal URL to Container URL after Activation" for the temporary solution.

8.4 Scaling Out Oracle Universal Content Management

Scaling out Oracle UCM involves two tasks:

- Creating a Common Location for the Oracle UCM Managed Servers
- Scaling Out the Oracle UCM Inbound Refinery Server

8.4.1 Creating a Common Location for the Oracle UCM Managed Servers

Do the following **prior** to performing the steps in Section 8.5, "Packing and Unpacking the Managed Server Domain Home":

1. Move the *CRMHOST1* `/u02/local/oracle/config/domains/`*CRMHOST1*`/CommonDomain/ucm/cs` directory from the local to the shared location: *ORACLE_BASE*`/config/domains/`*CRMHOST1*`/CommonDomain/ucm/cs`.
2. In the *ORACLE_BASE*`/config/domains/CRMHOST1/CommonDomain/ucm/cs/bin/intradoc.cfg` file, create the variables `intradocDir`, `vaultDir`, and `weblayoutDir` so that they point to the shared location, *ORACLE_BASE*`/config/domains/`*CRMHOST1*`/CommonDomain/ucm/cs`.

8.4.2 Scaling Out the Oracle UCM Inbound Refinery Server

You also must scale out the Oracle UCM Inbound Refinery server.

Note: `/u02/local/oracle/config/domains/`*`CRMHOST1`*`/CommonDomain/ucm/ibr` on *`CRMHOST1`* must already be on the local disk.

Do the following **after** performing the steps in Section 8.5, "Packing and Unpacking the Managed Server Domain Home" and Section 8.6, "Cloning Managed Servers and Assigning Them to CRMHOST2":

1. Edit the `/u02/local/oracle/config/domains/CRMHOST1/CommonDomain/ucm/ibr/config/config.cfg` file on *`CRMHOST1`* with the following:
   - `IDC_Name=CRMHOST1usoraclecom7012` (In this instance, *`CRMHOST1`* is **not** the fully qualified name.)
   - `InstanceMenuLabel=CRMHOST1usoraclecom7012`
   - `SocketHostAddressSecurityFilter=127.0.0.1|cluster node's addresses|`*`CRMHOST1`*` IP ADDRESS|`*`CRMHOST2`*` IP ADDRESS|0.0.0.0.0.0.0.1` (Include the cluster node's addresses.)
   - `HttpServerAddress=`*`CRMHOST1`*`:7012` (*`CRMHOST1`*'s server address instead of load balancer, as it is a singleton node.)
2. Start `UCM_server2`.
3. Navigate to `http://CRMHOST2:7012/ibr` (*`CRMHOST2`*'s `ibr`) and log in when prompted.

   The post-installation configuration page displays, as it is not a clustered node.
4. On the post-installation configuration page:
   a. Set the `intradoc` server port to **7035**.
   b. Ensure the port has both a unique IDC_Name and unique local paths on *`CRMHOST2`*.
   c. Set the Incoming Socket Connection Address Security Filter:

      ```
      SocketHostAddressSecurityFilter=127.0.0.1|
      cluster node's addresses|CRMHOST1
      IP ADDRESS|CRMHOST2
      IP ADDRESS|0.0.0.0.0.0.0.1
      ```

   d. Click **Submit**.
5. Navigate to `http://CRMHOST1:7012/cs` and select **Administration > providers**.

   ibrprovider is already defined in this list.
6. Add a new outgoing provider:
   a. Click **Add**.
   b. Set the values to match the values of **ibrprovider**.
   c. Set the HTTP host name to be second host address's *`CRMHOST2`* IP address.
7. Restart `UCM_server1` and `UCM_server2`.
8. Navigate to Inbound Refinery on second node: `http://CRMHOST2:7012/ibr`.
9. Select **Conversion Settings** and update the primary web rendition.

10. Navigate to the third-party application's **Settings > General OutsideIn Filter Options > Options**, and set the following path to the fonts: `/usr/share/X11/fonts/TTF`.

    Note: The font location can be specific to the operating system.

11. Restart `UCM_server2`.
12. Create a *COMMONUCMVH1* TCP VIP on the load-balancing router (LBR) with *CRMHOST1*`:7012` and *CRMHOST2*`:7012` as its members.

After scaling out the Oracle WebCenter managed server, do the following:

1. Log in to: `http://commoninternal.mycompany.com:7777/em`.
2. Navigate to **WebCenter > *WebCenter_managedserver* > WebCenter** (top left of right pane) **> Content repository > Oracle Content Server > Edit > RIDC Socket type** (in Connection details).
3. Select **Socket** and enter *COMMONUCMVH1* as the host.
4. Restart the Oracle WebCenter managed servers.

8.5 Packing and Unpacking the Managed Server Domain Home

Since the *CRMHOST1* domain directory file system is also available from *CRMHOST2*, both `pack` and `unpack` commands can be executed from *CRMHOST2*.

1. Do the following:
   a. Change directory to *ORACLE_BASE*`/products/fusionapps/oracle_common/common/bin`.
   b. Run the `pack` command:

```
CRMHOST2> ./pack.sh -managed=true -domain=ORACLE_BASE/config/domains/
CRMHOST1/CommonDomain -template=ORACLE_BASE/user_templates/
CommonDomain_managed.jar -template_name="Common_Managed_Server_Domain"
```

2. Ensure that `/u02/local/oracle/config/domains/`*CRMHOST2*`/CommonDomain` is empty, and then run the `unpack` command:

```
CRMHOST2> ./unpack.sh -domain=/u02/local/oracle/config/domains/CRMHOST2/
CommonDomain -template=ORACLE_BASE/user_templates/CommonDomain_managed.jar
```

   Here, *ORACLE_BASE* is shared, and `/u02/local` is local to *CRMHOST2*.

8.6 Cloning Managed Servers and Assigning Them to CRMHOST2

To add a managed server and assign it to *CRMHOST2*:

1. Log in to the Administration Server: `http://`*commoninternal*`.mycompany.com:7777/console`.
2. Navigate to **CommonDomain > Environment > Servers**.
3. Switch to **Lock & Edit** mode.
4. Select the ***Managed_Servers*** checkbox (for example, **HomePageServer_1**) and then click **Clone**.

5. Specify the following Server Identity attributes:
   - Server Name - `HomePageServer_2`

     Note: To ensure consistency in naming, copy the name of the server shown in **Server Identity** and paste it into the **Server Name** field. Then change the number to "_2".

   - Server Listen Address - <*CRMHOST2*>
   - Server Listen Port - leave "as is"
6. Click **OK**.

   You now should see the newly cloned sales server, `HomePageServer_2`.
7. Click **HomePageServer_2** and change the following attributes:
   - Machine - <*CRMHOST2*>
   - Cluster Name - Default, HomePageCluster
8. Click **Save** and then **Activate Changes**.
9. From the **Name** column, click the **HomepageServer_2** scaled-out server link.
10. Click **Lock & Edit**, and then choose the **Keystores** tab.
11. Ensure that the keystores value is **Custom Identity and Custom Trust**.
12. Do the following:
    a. Change the Custom Identity Keystore path to point to the *ORACLE_BASE*/products/fusionapps/wlserver_10.3/server/lib/*CRMHOST2*_fusion_identity.jks file.
    b. Leave the Custom Identity Keystore type blank.
    c. Change the Custom Identity Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."
    d. Re-enter the Confirm Custom Identity Keystore Passphrase.
    e. Ensure that the Confirm Custom Trust Keystore path is pointing to the *ORACLE_BASE*/products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks file.
    f. Leave the Custom Trust Keystore type blank.
    g. Change the Custom Trust Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."
    h. Re-enter the Custom Trust Keystore Passphrase.
    i. Click **Save**.
13. Choose the **SSL** tab.
    a. Make sure that Identity and Trust Locations is set to **Keystores**.
    b. Change the Private Key Alias to *CRMHOST2*_fusion.

c. Change the Private Key Passphrase to the *keypassword*, as described in the second bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

d. Re-enter the *keypassword* from Step c for the Confirm Private Key Passphrase.

e. Click **Save**.

14. Click **Activate Changes**.

15. Repeat Steps 2 to 14 for all the managed servers on this domain.

16. Set the following environment variable:

```
WLST_PROPERTIES="-Dweblogic.security.SSL.trustedCAKeyStore=ORACLE_BASE/
products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks"
```

17. Stop the domain's Administration Server:

```
CRMHOST1> ORACLE_BASE/config/domains/CRMHOST1/CommonDomain/bin/stopWebLogic.sh
```

18. Restart the domain's Administration Server:

```
CRMHOST2> ORACLE_BASE/products/fusionapps/wlserver_10.3/common/bin/wlst.sh

CRMHOST2> nmConnect(username='<username>', password='<password>',
domainName='CommonDomain', host='CRMHOST1',port='5556',
nmType='ssl', domainDir='ORACLE_BASE/config/domains/CRMHOST1/CommonDomain')

CRMHOST2> nmStart('AdminServer')
```

Note: The *username* and *password* used in the `nmConnect` are the Node Manager credentials (username and password) specified when creating the provisioning plan. This is shown in Figure 4–3 in "Using the Provisioning Process to Install Components for an Enterprise Deployment".

---

19. For the `wlcs_sipstate2` scaled-out server only, do the following:

a. Change directory to:

```
ORACLE_BASE/config/domains/CRMHOST1/CommonDomain/config/custom
```

b. In the `datatier.xml` file, after the following line:

```
<server-name>wlcs_sipstate1</server-name>
```

add

```
<server-name>wlcs_sipstate2<server-name>
```

20. Run the newly created managed server:

a. Navigate to **CommonDomain > Environment > Servers > Control**.

b. Select the newly created managed server and click **Start**.

c. Navigate to **CommonDomain > Environment > Servers** and check the **State** to verify that the newly created managed servers are running.

21. Log in to the Administration Server once again (`http://commoninternal.mycompany.com:7777/console`) and verify that all the managed servers, including scaled-out servers, are running.

Note: For all the scaled-up and scaled-out servers, change the Arguments in the `/u02/local/oracle/config/domains/HOSTNAME/DomainName/servers/ManagedServer/data/nodemanager/startup.properties` file to the following:

```
Arguments=-DJDBCProgramName\=DS/CommonDomain/HomePageServer_2
-Dserver.group\=HomePageCluster
```

Note: For all the scaled-up and scaled-out managed servers, do the following:

1. Access the Oracle WebLogic Server Administration Console for the CommonDomain:

   `http://commoninternal.mycompany.com:7777/console`

2. Navigate to **Environment > Servers** and click the "Managed Server" link.
3. First select the **Logging** tab and then the **HTTP** tab.
4. Update the following parameters:
   – Log file name: logs/access.log.%yyyyMMdd%
   – Rotation Type: By Time
   – Limit number of retained files: leave this option unchecked
   – Rotate log file on startup: leave this option unchecked
5. Click **Save**.
6. Expand **Advanced Node** and set the following:
   – Format: Extended
   – Extended Logging Format Fields:

   ```
   date time time-taken cs-method cs-uri sc-status
   sc(X-ORACLE-DMS-ECID) cs(ECID-Context) cs(Proxy-Remote-User)
   cs(Proxy-Client-IP)
   ```

7. Click **Save** and **Activate Changes**.
8. Restart the Managed Server for the changes to take affect.

8.7 Oracle HTTP Server Configuration

To configure Oracle HTTP Server:

1. On *WEBHOST1*:
   a. Change directory to *ORACLE_BASE*`/config/CommonDomain_webtier/config/OHS/ohs1/moduleconf`.
   b. Copy `FusionVirtualHost_fs.conf` to `FusionVirtualHost_fs.conf.org`.

2. Edit the `FusionVirtualHost_fs.conf` file, adding the scaled-out host and port to all the WebLogic Application Clusters. Example 8–1 shows sample code for HomePageServer.

Example 8–1 Sample "HomePageServer" Code

```
<Location /HomePage>
    SetHandler weblogic-handler
    WebLogicCluster <CRMHOST1:port>,<CRMHOST2:port>
</Location>
```

3. Repeat Step 2 for all applications.
4. Restart Oracle HTTP Server: `cd` to *`ORACLE_BASE`*`/config/CommonDomain_webtier/bin` and enter the following:

```
WEBHOST1> ./opmnctl stopall
WEBHOST1> ./opmnctl startall
```

5. Repeat Steps 1 through 4 on *`WEBHOST2`*.

8.8 Validating the System

You should verify URLs to ensure that the appropriate routing and failover are working.

To verify the URLs:

1. Log in to the `CommonDomain` Oracle WebLogic Server Administration Console and stop all the managed servers on the *`CRMHOST1`* while the managed servers on *`CRMHOST2`* are running.
2. Access the following URLs to verify that routing and failover are functioning properly. (Ensure the log in prompt is visible.)
   - `https://commonexternal.mycompany.com/helpPortal/faces/AtkHelpPortalMain`
   - `https://commonexternal.mycompany.com/homePage/faces/AtkHomePageWelcome`
3. Log in to the `CommonDomain` Oracle WebLogic Server Administration Console and stop all the managed servers on *`CRMHOST2`*.
4. Start the managed servers on *`CRMHOST1`*.
5. Repeat Step 2. (Ensure the log in prompt is visible.)
6. Start all the managed servers on *`CRMHOST2`* and verify that they are running on *`CRMHOST1`*.

9

Scaling Out the Oracle Fusion Human Capital Management Domain

This chapter describes how to scale out the Oracle Fusion Human Capital Management domain.

This chapter includes the following topics:

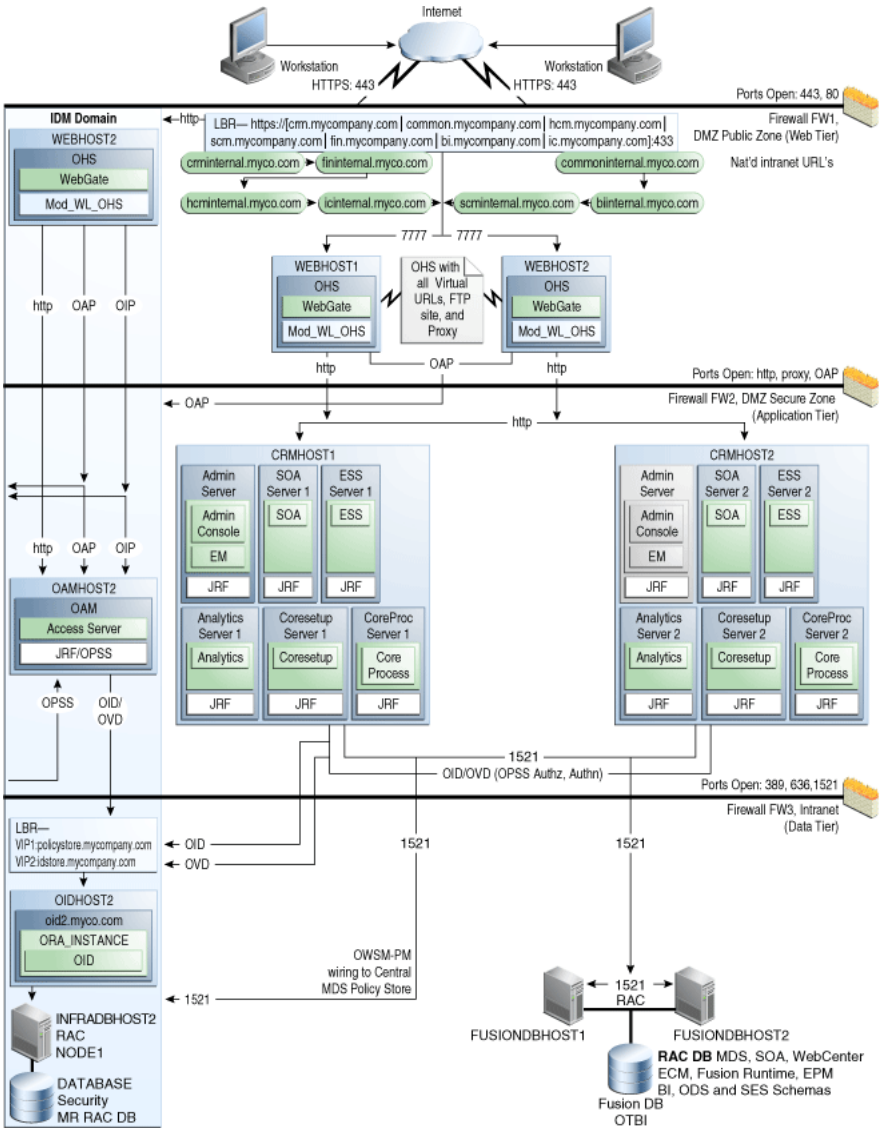
- Section 9.1, "Overview of the Oracle Fusion Human Capital Management Domain"
- Section 9.2, "Prerequisites for Scaling Out the Oracle Fusion Human Capital Management Domain"
- Section 9.3, "Adding a New Machine in the Oracle WebLogic Server Console"
- Section 9.4, "Packing and Unpacking the Managed Server Domain Home"
- Section 9.5, "Cloning Managed Servers and Assigning Them to CRMHOST2"
- Section 9.6, "Oracle HTTP Server Configuration"
- Section 9.7, "Validating the System"

9.1 Overview of the Oracle Fusion Human Capital Management Domain

The Oracle Fusion Human Capital Management domain provides the user-management flow needed to create a user, which is executed via Oracle Fusion Human Capital Management/Oracle Identity Management integration. The flow is available as part of Customer Data Management.

Figure 9–1 shows the Oracle Fusion Human Capital Management domain within the overall reference enterprise deployment topology for Oracle Fusion Applications.

Figure 9–1 Reference Topology for the Oracle Fusion Human Capital Management Domain



9.2 Prerequisites for Scaling Out the Oracle Fusion Human Capital Management Domain

Before you begin, ensure the following:

- Node Manager has been started in the Secure Sockets Layer (SSL) mode by following the instructions in Chapter 6, "Configuring Node Manager"
- You are starting with a clean machine if it is the first time it is being used for a scale out
- The `/etc/hosts` file has proper entries. To verify, ping this machine with the fully qualified name of the machine
- The user created on *CRMHOST2* should the same as the user on *CRMHOST1*
- The directory structure `/u01/oracle` is mounted to same shared file system as *CRMHOST1*

- The directory structure `/u02/local/oracle/config` on *CRMHOST2* has been created
- The initial Oracle Fusion Customer Relationship Management deployment on *CRMHOST1* has already been done and verified by provisioning

9.3 Adding a New Machine in the Oracle WebLogic Server Console

To add a new machine:

1. Log in to the Administration Server: `http://`*hcminternal*`.mycompany.com:7777/console`.
2. Navigate to **HCMDomain > Environment > Machines**.

   LocalMachine is located in the right-hand pane.
3. In the left-hand pane, click **Lock & Edit**.
4. In the right-hand pane, first click **New** to add the remote machine, and then specify the following:
   - Name - enter *CRMHOST2*
   - Machine operating system - Unix
5. Click **Next**.
6. In the window that opens, set the following attributes:
   - Type - SSL
   - Listen Address - <*CRMHOST2*>

     Note: The "localhost" default value here is wrong.
   - Listen port - 5556
7. Click **Finish** and activate the changes.

   Note: If you get an error when activating the changes, see Section 18.8.18, "Administration Console Redirects from Internal URL to Container URL after Activation" for the temporary solution.

9.4 Packing and Unpacking the Managed Server Domain Home

Since the *CRMHOST1* domain directory file system is also available from *CRMHOST2*, both `pack` and `unpack` commands can be executed from *CRMHOST2*.

1. Do the following:
   a. Change directory to *ORACLE_BASE*`/products/fusionapps/oracle_common/common/bin`.
   b. Run the `pack` command:

```
CRMHOST2> ./pack.sh -managed=true -domain=ORACLE_BASE/config/domains/
CRMHOST1/HCMDomain -template=ORACLE_BASE/user_templates/
HCMDomain_managed.jar -template_name="HCM_Managed_Server_Domain"
```

2. Ensure that `/u02/local/oracle/config/domains/CRMHOST2/HCMDomain` is empty, and then run the `unpack` command:

```
CRMHOST2> ./unpack.sh -domain=/u02/local/oracle/config/domains/
CRMHOST2/HCMDomain -template=ORACLE_BASE/user_templates/HCMDomain_managed.jar
```

   Here, *ORACLE_BASE* is shared, and `/u02/local` is local to *CRMHOST2*.

9.5 Cloning Managed Servers and Assigning Them to CRMHOST2

To add a managed server and assign it to *CRMHOST2*:

1. Log in to the Administration Server: `http://hcminternal.mycompany.com:7777/console`
2. Navigate to **HCMDomain > Environment > Servers**.
3. Switch to **Lock & Edit** mode.
4. Select the ***Managed_Servers*** checkbox (for example, **CoreSetupServer_1**) and then click **Clone**.
5. Specify the following Server Identity attributes:
   - Server Name - `CoreSetupServer_2`

     Note: To ensure consistency in naming, copy the name of the server shown in **Server Identity** and paste it into the **Server Name** field. Then change the number to "_2".

   - Server Listen Address - <*CRMHOST2*>
   - Server Listen Port - leave "as is"
6. Click **OK**.

   You now should see the newly cloned sales server, `CoreSetupServer_2`.
7. Click **CoreSetupServer_2** and change the following attributes:
   - Machine - <*CRMHOST2*>
   - Cluster Name - Default, CoreSetupCluster
8. Click **Save** and then **Activate Changes**.
9. From the **Name** column, click the **CoreSetupServer_2** scaled-out server link.
10. Click **Lock & Edit**, and then choose the **Keystores** tab.
11. Ensure that the keystores value is **Custom Identity and Custom Trust**.
12. Do the following:
    a. Change the Custom Identity Keystore path to point to the *ORACLE_BASE*/`products/fusionapps/wlserver_10.3/server/lib/`*CRMHOST2*`_fusion_identity.jks` file.
    b. Leave the Custom Identity Keystore type blank.

c. Change the Custom Identity Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

d. Re-enter the Confirm Custom Identity Keystore Passphrase.

e. Ensure that the Confirm Custom Trust Keystore path is pointing to the *ORACLE_BASE*/products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks file.

f. Leave the Custom Trust Keystore type blank.

g. Change the Custom Trust Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

h. Re-enter the Custom Trust Keystore Passphrase.

i. Click **Save**.

13. Choose the **SSL** tab.

a. Make sure that Identity and Trust Locations is set to **Keystores**.

b. Change the Private Key Alias to *CRMHOST2*_fusion.

c. Change the Private Key Passphrase to the *keypassword*, as described in the second bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

d. Re-enter the *keypassword* from Step c for the Confirm Private Key Passphrase.

e. Click **Save**.

14. Click **Activate Changes**.

15. Repeat Steps 2 to 14 for all the managed servers on this domain.

16. Set the following environment variable:

```
WLST_PROPERTIES="-Dweblogic.security.SSL.trustedCAKeyStore=ORACLE_BASE/
products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks"
```

17. Stop the domain's Administration Server:

```
CRMHOST1> ORACLE_BASE/config/domains/CRMHOST1/HCMDomain/bin/stopWebLogic.sh
```

18. Restart the domain's Administration Server:

```
CRMHOST2> ORACLE_BASE/products/fusionapps/wlserver_10.3/common/bin/wlst.sh

CRMHOST2> nmConnect(username='<username>', password='<password>',
domainName='HCMDomain', host='CRMHOST1',port='5556',
nmType='ssl', domainDir='ORACLE_BASE/config/domains/CRMHOST1/HCMDomain')

CRMHOST2> nmStart('AdminServer')
```

> **Note:** The *username* and *password* used in the `nmConnect` are the Node Manager credentials (username and password) specified when creating the provisioning plan. This is shown in Figure 4–3 in "Using the Provisioning Process to Install Components for an Enterprise Deployment".

19. Run the newly created managed server:

a. Navigate to **HCMDomain > Environment > Servers > Control**.

b. Select the newly created managed server and click **Start**.

c. Navigate to **HCMDomain > Environment > Servers** and check the **State** to verify that the newly created managed servers are running.

20. Log in to the Administration Server once again (`http://hcminternal.mycompany.com:7777/console`) and verify that all the managed servers, including scaled-out servers, are running.

Note: For all the scaled-up and scaled-out servers, change the Arguments in the `/u02/local/oracle/config/domains/HOSTNAME/DomainName/servers/ManagedServer/data/nodemanager/startup.properties` file to the following:

```
Arguments=-DJDBCProgramName\=DS/HCMDomain/CoreSetupServer_2
-Dserver.group\=CoreSetupCluster
```

Note: For all the scaled-up and scaled-out managed servers, do the following:

1. Access the Oracle WebLogic Server Administration Console for the HCMDomain:

   `http://hcminternal.mycompany.com:7777/console`

2. Navigate to **Environment > Servers** and click the "Managed Server" link.
3. First select the **Logging** tab and then the **HTTP** tab.
4. Update the following parameters:

   – Log file name: logs/access.log.%yyyyMMdd%

   – Rotation Type: By Time

   – Limit number of retained files: leave this option unchecked

   – Rotate log file on startup: leave this option unchecked

5. Click **Save**.
6. Expand **Advanced Node** and set the following:

   – Format: Extended

   – Extended Logging Format Fields:

   ```
   date time time-taken cs-method cs-uri sc-status
   sc(X-ORACLE-DMS-ECID) cs(ECID-Context) cs(Proxy-Remote-User)
   cs(Proxy-Client-IP)
   ```

7. Click **Save** and **Activate Changes**.
8. Restart the Managed Server for the changes to take affect.

9.6 Oracle HTTP Server Configuration

To configure Oracle HTTP Server:

1. On *WEBHOST1*:

   a. Change directory to *ORACLE_BASE*`/config/CommonDomain_webtier/config/OHS/ohs1/moduleconf`.

   b. Copy `FusionVirtualHost_hcm.conf` to `FusionVirtualHost_hcm.conf.org`.

2. Edit the `FusionVirtualHost_hcm.conf` file, adding the scaled-out host and port to all the WebLogic Application Clusters. Example 9–1 shows sample code for CoreSetupServer.

Example 9–1 Sample "CoreSetupServer" Code

```
<Location /hcmCoreSetup>
    SetHandler weblogic-handler
    WebLogicCluster <CRMHOST1:port>,<CRMHOST2:port>
</Location>
```

3. Repeat Step 2 for all applications.
4. Restart Oracle HTTP Server: `cd` to *`ORACLE_BASE`*`/config/CommonDomain_webtier/bin` and enter the following:

```
WEBHOST1> ./opmnctl stopall
WEBHOST1> ./opmnctl startall
```

5. Repeat Steps 1 through 4 on *`WEBHOST2`*.

9.7 Validating the System

You should verify URLs to ensure that the appropriate routing and failover are working.

To verify the URLs:

1. Log in to the `HCMDomain` Oracle WebLogic Server Administration Console and stop all the managed servers on the *`CRMHOST1`* while the managed servers on *`CRMHOST2`* are running.
2. Access the following URL to verify that routing and failover are functioning properly. (Ensure the log in prompt is visible.)
   - `https://hcmexternal.mycompany.com/hcmCoreSetup/faces/HcmWSIntWA`
3. Log in to the `HCMDomain` Oracle WebLogic Server Administration Console and stop all the managed servers on *`CRMHOST2`*.
4. Start the managed servers on *`CRMHOST1`*.
5. Repeat Step 2. (Ensure the log in prompt is visible.)
6. Start all the managed servers on *`CRMHOST2`* and verify that they are running on *`CRMHOST1`*.

10

Scaling Out the Oracle Fusion Supply Chain Management Domain

This chapter describes how to scale out the Oracle Fusion Supply Chain Management domain.

This chapter includes the following topics:

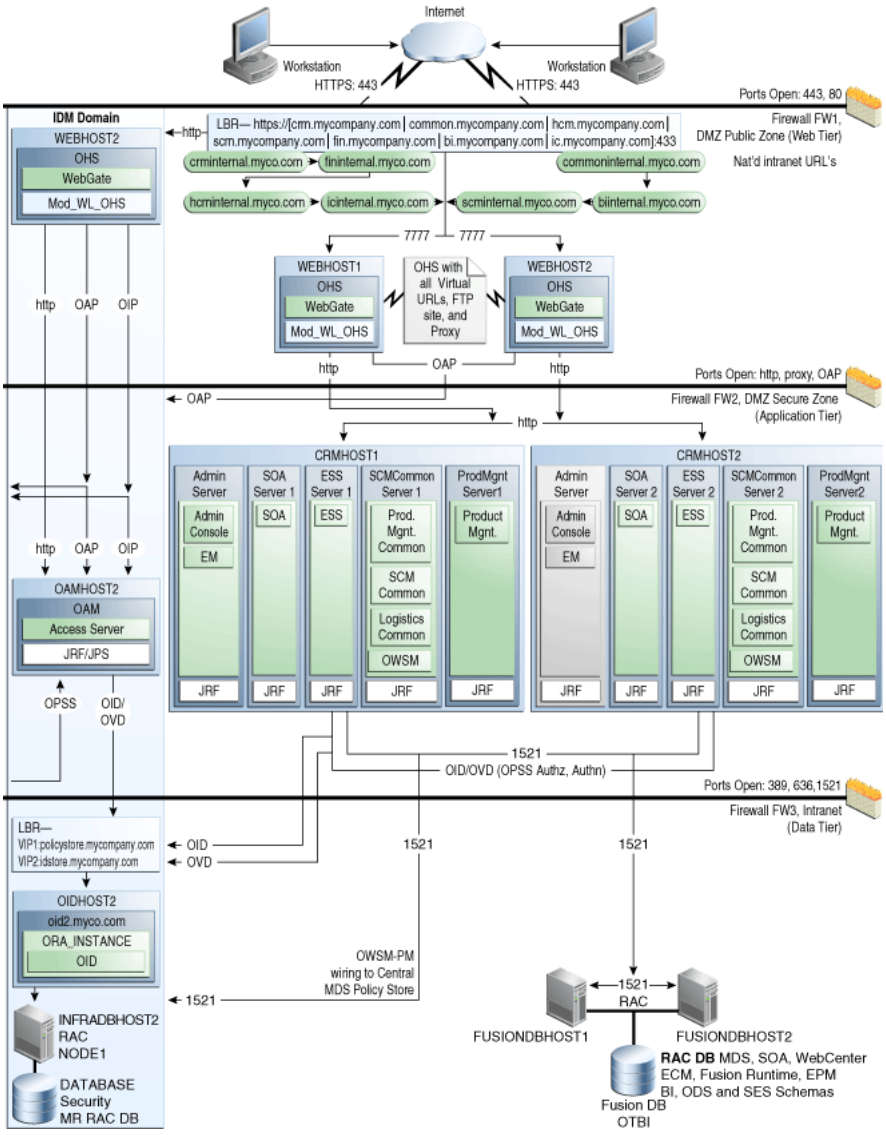
- Section 10.1, "Overview of the Oracle Fusion Supply Chain Management Domain"
- Section 10.2, "Prerequisites for Scaling Out the Oracle Fusion Supply Chain Management Domain"
- Section 10.3, "Adding a New Machine in the Oracle WebLogic Server Console"
- Section 10.4, "Packing and Unpacking the Managed Server Domain Home"
- Section 10.5, "Cloning Managed Servers and Assigning Them to CRMHOST2"
- Section 10.6, "Oracle HTTP Server Configuration"
- Section 10.7, "Validating the System"

10.1 Overview of the Oracle Fusion Supply Chain Management Domain

Oracle Fusion Customer Relationship Management uses Oracle Fusion Supply Chain Management for products and product groups. The Oracle Fusion Supply Chain Management domain provides the flows to import any existing customer product and product catalog into Oracle Fusion Customer Relationship Management.

Figure 10–1 shows the Oracle Fusion Supply Chain Management domain within the overall reference enterprise deployment topology for Oracle Fusion Applications.

Figure 10–1 Reference Topology for Oracle Fusion Supply Chain Management



10.2 Prerequisites for Scaling Out the Oracle Fusion Supply Chain Management Domain

Before you begin, ensure the following:

- Node Manager has been started in the Secure Sockets Layer (SSL) mode by following the instructions in Chapter 6, "Configuring Node Manager"
- You are starting with a clean machine if it is the first time it is being used for a scale out
- The `/etc/hosts` file has proper entries. To verify, ping this machine with the fully qualified name of the machine
- The user created on *CRMHOST2* should the same as the user on *CRMHOST1*
- The directory structure `/u01/oracle` is mounted to same shared file system as *CRMHOST1*

- The directory structure `/u02/local/oracle/config` on *CRMHOST2* has been created
- The initial Oracle Fusion Customer Relationship Management deployment on *CRMHOST1* has already been done and verified by provisioning

10.3 Adding a New Machine in the Oracle WebLogic Server Console

To add a new machine:

1. Log in to the Administration Server: `http://`*scminternal*`.mycompany.com:7777/console`.
2. Navigate to **SCMDomain > Environment > Machines**.

   Local Machine is located in the right-hand pane.
3. In the left-hand pane, click **Lock & Edit**.
4. In the right-hand pane, first click **New** to add the remote machine, and then specify the following:
   - Name - enter *CRMHOST2*
   - Machine operating system - Unix
5. Click **Next**.
6. In the window that opens, set the following attributes:
   - Type - SSL
   - Listen Address - <*CRMHOST2*>

     Note: The "localhost" default value here is wrong.
   - Listen port - 5556
7. Click **Finish** and activate the changes.

Note: If you get an error when activating the changes, see Section 18.8.18, "Administration Console Redirects from Internal URL to Container URL after Activation" for the temporary solution.

10.4 Packing and Unpacking the Managed Server Domain Home

Since the *CRMHOST1* domain directory file system is also available from *CRMHOST2*, both the `pack` and `unpack` commands can be executed from the *CRMHOST2*.

1. Do the following:
   a. Change directory to *ORACLE_BASE*`/products/fusionapps/oracle_common/common/bin`.
   b. Run the `pack` command:

      ```
      CRMHOST2> ./pack.sh -managed=true -domain=ORACLE_BASE/config/domains/
      ```

```
CRMHOST1/SCMDomain -template=ORACLE_BASE/user_templates/
SCMDomain_managed.jar -template_name="SCM_Managed_Server_Domain"
```

2. Ensure that `/u02/local/oracle/config/domains/CRMHOST2/SCMDomain` is empty, and then run the `unpack` command:

```
CRMHOST2> ./unpack.sh -domain=/u02/local/oracle/config/domains/CRMHOST2/
SCMDomain -template=ORACLE_BASE/user_templates/SCMDomain_managed.jar
```

   Here, *ORACLE_BASE* is shared, and `/u02/local` is local to *CRMHOST2*.

10.5 Cloning Managed Servers and Assigning Them to CRMHOST2

To add a managed server and assign it to *CRMHOST2*:

1. Log in to the Administration Server: `http://scminternal.mycompany.com:7777/console`.
2. Navigate to **SCMDomain > Environment > Servers**.
3. Switch to **Lock & Edit** mode.
4. Select the ***Managed_Servers*** checkbox (for example, **ProductManagementServer_1**) and then click **Clone**.
5. Specify the following Server Identity attributes:
   - Server Name - `ProductManagementServer_2`

     Note: To ensure consistency in naming, copy the name of the server shown in **Server Identity** and paste it into the **Server Name** field. Then change the number to "_2".

   - Server Listen Address - <*CRMHOST2*>
   - Server Listen Port - leave "as is"
6. Click **OK**.

   You now should see the newly cloned sales server, `ProductManagementServer_2`.
7. Click **ProductManagementServer_2** and change the following attributes:
   - Machine - <*CRMHOST2*>
   - Cluster Name - Default, ProductManagementCluster
8. Click **Save** and then **Activate Changes**.
9. From the **Name** column, click the **ProductManagementServer_2** scaled-out server link.
10. Click **Lock & Edit**, and then choose the **Keystores** tab.
11. Ensure that the keystores value is **Custom Identity and Custom Trust**.
12. Do the following:
    a. Change the Custom Identity Keystore path to point to the *ORACLE_BASE*/`products/fusionapps/wlserver_10.3/server/lib/CRMHOST2_fusion_identity.jks` file.
    b. Leave the Custom Identity Keystore type blank.

c. Change the Custom Identity Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

d. Re-enter the Confirm Custom Identity Keystore Passphrase.

e. Ensure that the Confirm Custom Trust Keystore path is pointing to the *ORACLE_BASE*/products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks file.

f. Leave the Custom Trust Keystore type blank.

g. Change the Custom Trust Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

h. Re-enter the Custom Trust Keystore Passphrase.

i. Click **Save**.

13. Choose the **SSL** tab.

a. Make sure that Identity and Trust Locations is set to **Keystores**.

b. Change the Private Key Alias to *CRMHOST2*_fusion.

c. Change the Private Key Passphrase to the *keypassword*, as described in the second bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

d. Re-enter the *keypassword* from Step c for the Confirm Private Key Passphrase.

e. Click **Save**.

14. Click **Activate Changes**.

15. Repeat Steps 2 to 14 for all the managed servers on this domain.

16. Set the following environment variable:

```
WLST_PROPERTIES="-Dweblogic.security.SSL.trustedCAKeyStore=ORACLE_BASE/
products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks"
```

17. Stop the domain's Administration Server:

```
CRMHOST1> ORACLE_BASE/config/domains/CRMHOST1/SCMDomain/bin/stopWebLogic.sh
```

18. Restart the domain's Administration Server:

```
CRMHOST2> ORACLE_BASE/products/fusionapps/wlserver_10.3/common/bin/wlst.sh

CRMHOST2> nmConnect(username='<username>', password='<password>',
domainName='SCMDomain', host='CRMHOST1',port='5556',
nmType='ssl', domainDir='ORACLE_BASE/config/domains/CRMHOST1/SCMDomain')

CRMHOST2> nmStart('AdminServer')
```

> **Note:** The *username* and *password* used in the `nmConnect` are the Node Manager credentials (username and password) specified when creating the provisioning plan. This is shown in Figure 4–3 in "Using the Provisioning Process to Install Components for an Enterprise Deployment".

19. Run the newly created managed server:

a. Navigate to **SCMDomain > Environment > Servers > Control**.

b. Select the newly created managed server and click **Start**.

c. Navigate to **SCMDomain > Environment > Servers** and check the **State** to verify that the newly created managed servers are running.

20. Log in to the Administration Server once again (`http://scminternal.mycompany.com:7777/console`) and verify that all the managed servers, including scaled-out servers, are running.

Note: For all the scaled-up and scaled-out servers, change the Arguments in the `/u02/local/oracle/config/domains/HOSTNAME/DomainName/servers/ManagedServer/data/nodemanager/startup.properties` file to the following:

```
Arguments=-DJDBCProgramName\=DS/SCMDomain/
ProductManagementServer_2 -Dserver.group\=ProductManagementCluster
```

Note: For all the scaled-up and scaled-out managed servers, do the following:

1. Access the Oracle WebLogic Server Administration Console for the SCMDomain:

   `http://scminternal.mycompany.com:7777/console`

2. Navigate to **Environment > Servers** and click the "Managed Server" link.
3. First select the **Logging** tab and then the **HTTP** tab.
4. Update the following parameters:

   – Log file name: logs/access.log.%yyyyMMdd%

   – Rotation Type: By Time

   – Limit number of retained files: leave this option unchecked

   – Rotate log file on startup: leave this option unchecked

5. Click **Save**.
6. Expand **Advanced Node** and set the following:

   – Format: Extended

   – Extended Logging Format Fields:

   ```
   date time time-taken cs-method cs-uri sc-status
   sc(X-ORACLE-DMS-ECID) cs(ECID-Context) cs(Proxy-Remote-User)
   cs(Proxy-Client-IP)
   ```

7. Click **Save** and **Activate Changes**.
8. Restart the Managed Server for the changes to take affect.

10.6 Oracle HTTP Server Configuration

To configure Oracle HTTP Server:

1. On *WEBHOST1*:

   a. Change directory to *ORACLE_BASE*`/config/CommonDomain_webtier/config/OHS/ohs1/moduleconf`.

b. Copy `FusionVirtualHost_scm.conf` to `FusionVirtualHost_scm.conf.org`.

2. Edit the `FusionVirtualHost_scm.conf` file, adding the scaled-out host and port to all the WebLogic Application Clusters. Example 10–1 shows sample code for ProductManagementServer.

Example 10–1 Sample "productManagementServer" Code

```
<Location /productManagement>
    SetHandler weblogic-handler
    WebLogicCluster <CRMHOST1:port>,<CRMHOST2:port>
</Location>
```

3. Repeat Step 2 for all applications.

4. Restart Oracle HTTP Server: `cd` to *`ORACLE_BASE`*`/config/CommonDomain_webtier/bin` and enter the following:

```
WEBHOST1> ./opmnctl stopall
WEBHOST1> ./opmnctl startall
```

5. Repeat Steps 1 through 4 on *`WEBHOST2`*.

10.7 Validating the System

You should verify URLs to ensure that the appropriate routing and failover are working.

To verify the URLs:

1. Log in to the `SCMDomain` Oracle WebLogic Server Administration Console and stop all the managed servers on the *`CRMHOST1`* while the managed servers on *`CRMHOST2`* are running.

2. Access the following URL to verify that routing and failover are functioning properly. (Ensure the log in prompt is visible.)

   - `https://scmexternal.mycompany.com/productManagement/faces/ItemDashboard`

3. Log in to the `SCMDomain` Oracle WebLogic Server Administration Console and stop all the managed servers on *`CRMHOST2`*.

4. Start the managed servers on *`CRMHOST1`*.

5. Repeat Step 2. (Ensure the log in prompt is visible.)

6. Start all the managed servers on *`CRMHOST2`* and verify that they are running on *`CRMHOST1`*.

11

Scaling Out the Oracle Fusion Financials Domain

This chapter describes how to scale out the Oracle Fusion Financials domain.

This chapter includes the following topics:

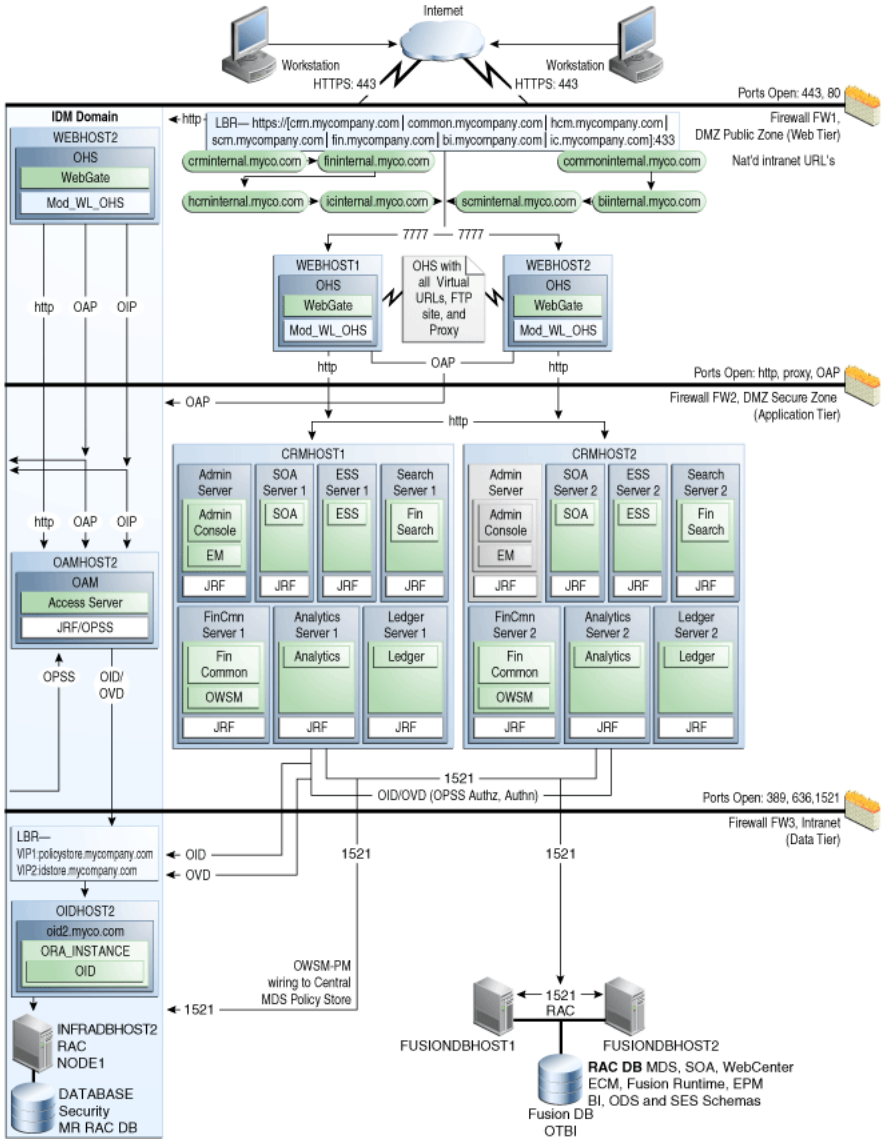
- Section 11.1, "Overview of the Oracle Fusion Financials Domain"
- Section 11.2, "Prerequisites for Scaling Out the Oracle Fusion Financials Domain"
- Section 11.3, "Adding a New Machine in the Oracle WebLogic Server Console"
- Section 11.4, "Packing and Unpacking the Managed Server Domain Home"
- Section 11.5, "Cloning Managed Servers and Assigning Them to CRMHOST2"
- Section 11.6, "Oracle HTTP Server Configuration"
- Section 11.7, "Validating the System"

11.1 Overview of the Oracle Fusion Financials Domain

The Oracle Financials domain provides Oracle Fusion Customer Relationship Management with Currency, Calendar, and other functionality. The Oracle Fusion Financials domain exposes setup portlets that the Oracle Fusion Customer Relationship Management implementation manager can use via Functional Setup Manager.

Figure 11–1 shows the Oracle Fusion Financials domain within the overall reference enterprise deployment topology for Oracle Fusion Applications.

Figure 11–1 Reference Topology for Oracle Fusion Financials Domain



11.2 Prerequisites for Scaling Out the Oracle Fusion Financials Domain

Before you begin, ensure the following:

- Node Manager has been started in the Secure Sockets Layer (SSL) mode by following the instructions in Chapter 6, "Configuring Node Manager"
- You are starting with a clean machine if it is the first time it is being used for a scale out
- The `/etc/hosts` file has proper entries. To verify, ping this machine with the fully qualified name of the machine
- The user created on *CRMHOST2* should the same as the user on *CRMHOST1*
- The directory structure `/u01/oracle` is mounted to same shared file system as *CRMHOST1*
- The directory structure `/u02/local/oracle/config` on *CRMHOST2* has been created

- The initial Oracle Fusion Customer Relationship Management deployment on *CRMHOST1* has already been done and verified by provisioning

11.3 Adding a New Machine in the Oracle WebLogic Server Console

To add a new machine:

1. Log in to the Administration Server: `http://`*`fininternal`*`.mycompany.com:7777/console`.
2. Navigate to **FinancialDomain > Environment > Machines**.

   LocalMachine is located in the right-hand pane.
3. In the left-hand pane, click **Lock & Edit**.
4. In the right-hand pane, first click **New** to add the remote machine, and then specify the following:
   - Name - enter *CRMHOST2*
   - Machine operating system - Unix
5. Click **Next**.
6. In the window that opens, set the following attributes:
   - Type - SSL
   - Listen Address - <*CRMHOST2*>

     Note: The "localhost" default value here is wrong.
   - Listen port - 5556
7. Click **Finish** and activate the changes.

Note: If you get an error when activating the changes, see Section 18.8.18, "Administration Console Redirects from Internal URL to Container URL after Activation" for the temporary solution.

11.4 Packing and Unpacking the Managed Server Domain Home

Since the *CRMHOST1* domain directory file system is also available from *CRMHOST2*, both the `pack` and `unpack` commands can be executed from the *CRMHOST2*.

1. Do the following:
   a. Change directory to *ORACLE_BASE*`/products/fusionapps/oracle_common/common/bin`.
   b. Run the `pack` command:

```
CRMHOST2> ./pack.sh -managed=true -domain=ORACLE_BASE/config/domains/
CRMHOST1/FinancialDomain -template=ORACLE_BASE/user_templates/
FinancialDomain_managed.jar -template_name=
"Financial_Managed_Server_Domain"
```

2. Ensure that `/u02/local/oracle/config/domains/CRMHOST2/FinancialDomain` is empty, and then run the `unpack` command:

   ```
   CRMHOST2> ./unpack.sh -domain=/u02/local/oracle/config/domains/CRMHOST2/
   FinancialDomain -template=ORACLE_BASE/user_templates/
   FinancialDomain_managed.jar
   ```

   Here, *ORACLE_BASE* is shared, and `/u02/local` is local to *CRMHOST2*.

11.5 Cloning Managed Servers and Assigning Them to CRMHOST2

To add a managed server and assign it to *CRMHOST2*:

1. Log in to the Administration Server: `http://`*`fininternal`*`.mycompany.com:7777/console`.
2. Navigate to **FinancialDomain > Environment > Servers**.
3. Switch to **Lock & Edit** mode.
4. Select the ***Managed_Servers*** checkbox (for example, **GeneralLedgerServer_1**) and then click **Clone**.
5. Specify the following Server Identity attributes:
   - Server Name - `GeneralLedgerServer_2`

     Note: To ensure consistency in naming, copy the name of the server shown in **Server Identity** and paste it into the **Server Name** field. Then change the number to "_2".

   - Server Listen Address - <*CRMHOST2*>
   - Server Listen Port - leave "as is"
6. Click **OK**.

   You now should see the newly cloned server, `GeneralLedgerServer_2`.
7. Click **GeneralLedgerServer_2** and change the following attributes:
   - Machine - <*CRMHOST2*>
   - Cluster Name - Default, GeneralLedgerCluster
8. Click **Save** and then **Activate Changes**.
9. From the **Name** column, click the **GeneralLedgerServer_2** scaled-out server link.
10. Click **Lock & Edit**, and then choose the **Keystores** tab.
11. Ensure that the keystores value is **Custom Identity and Custom Trust**.
12. Do the following:
    a. Change the Custom Identity Keystore path to point to the *ORACLE_BASE*`/products/fusionapps/wlserver_10.3/server/lib/`*CRMHOST2*`_fusion_identity.jks` file.
    b. Leave the Custom Identity Keystore type blank.
    c. Change the Custom Identity Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

d. Re-enter the Confirm Custom Identity Keystore Passphrase.

e. Ensure that the Confirm Custom Trust Keystore path is pointing to the *ORACLE_BASE*/products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks file.

f. Leave the Custom Trust Keystore type blank.

g. Change the Custom Trust Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

h. Re-enter the Custom Trust Keystore Passphrase.

i. Click **Save**.

13. Choose the **SSL** tab.

a. Make sure that Identity and Trust Locations is set to **Keystores**.

b. Change the Private Key Alias to *CRMHOST2*_fusion.

c. Change the Private Key Passphrase to the *keypassword*, as described in the second bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

d. Re-enter the *keypassword* from Step c for the Confirm Private Key Passphrase.

e. Click **Save**.

14. Click **Activate Changes**.

15. Repeat Steps 2 to 14 for all the managed servers on this domain.

16. Set the following environment variable:

```
WLST_PROPERTIES="-Dweblogic.security.SSL.trustedCAKeyStore=ORACLE_BASE/
products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks"
```

17. Stop the domain's Administration Server:

```
CRMHOST1> ORACLE_BASE/config/domains/CRMHOST1/FinancialDomain/bin/
stopWebLogic.sh
```

18. Restart the domain's Administration Server:

```
CRMHOST2> ORACLE_BASE/products/fusionapps/wlserver_10.3/common/bin/wlst.sh

CRMHOST2> nmConnect(username='<username>', password='<password>',
domainName='FinancialDomain', host='CRMHOST1',port='5556',
nmType='ssl', domainDir='ORACLE_BASE/config/domains/CRMHOST1/FinancialDomain')

CRMHOST2> nmStart('AdminServer')
```

> **Note:** The *username* and *password* used in the nmConnect are the Node Manager credentials (username and password) specified when creating the provisioning plan. This is shown in Figure 4–3 in "Using the Provisioning Process to Install Components for an Enterprise Deployment".

19. Run the newly created managed server:

a. Navigate to **FinancialDomain > Environment > Servers > Control**.

b. Select the newly created managed server and click **Start**.

c. Navigate to **FinancialDomain > Environment > Servers** and check the **State** to verify that the newly created managed servers are running.

20. Log in to the Administration Server once again (`http://`*`fininternal`*`.mycompany.com:7777/console`) and verify that all the managed servers, including scaled-out servers, are running.

Note: For all the scaled-up and scaled-out servers, change the Arguments in the `/u02/local/oracle/config/domains/`*`HOSTNAME`*`/`*`DomainName`*`/servers/`*`ManagedServer`*`/data/nodemanager/startup.properties` file to the following:

```
Arguments=-DJDBCProgramName\=DS/FinancialDomain/
GeneralLedgerServer_2 -Dserver.group\=GeneralLedgerCluster
```

Note: For all the scaled-up and scaled-out managed servers, do the following:

1. Access the Oracle WebLogic Server Administration Console for the FinancialDomain:

   `http://`*`fininternal`*`.mycompany.com:7777/console`

2. Navigate to **Environment > Servers** and click the "Managed Server" link.
3. First select the **Logging** tab and then the **HTTP** tab.
4. Update the following parameters:

   – Log file name: logs/access.log.%yyyyMMdd%

   – Rotation Type: By Time

   – Limit number of retained files: leave this option unchecked

   – Rotate log file on startup: leave this option unchecked

5. Click **Save**.
6. Expand **Advanced Node** and set the following:

   – Format: Extended

   – Extended Logging Format Fields:

   ```
   date time time-taken cs-method cs-uri sc-status
   sc(X-ORACLE-DMS-ECID) cs(ECID-Context) cs(Proxy-Remote-User)
   cs(Proxy-Client-IP)
   ```

7. Click **Save** and **Activate Changes**.
8. Restart the Managed Server for the changes to take affect.

11.6 Oracle HTTP Server Configuration

To configure Oracle HTTP Server:

1. On *`WEBHOST1`*:

   a. Change directory to *`ORACLE_BASE`*`/config/CommonDomain_webtier/config/OHS/ohs1/moduleconf`.

   b. Copy `FusionVirtualHost_fin.conf` to `FusionVirtualHost_fin.conf.org`.

2. Edit the `FusionVirtualHost_fin.conf` file, adding the scaled-out host and port to all the WebLogic Application Clusters. Example 11–1 shows sample code for GeneralLedgerServer.

Example 11–1 Sample "GeneralLedgerServer" Code

```
<Location /GeneralLedger>
    SetHandler weblogic-handler
    WebLogicCluster <CRMHOST1:port>,<CRMHOST2:port>
</Location>
```

3. Repeat Step 2 for all applications.
4. Restart Oracle HTTP Server: `cd` to *`ORACLE_BASE`*`/config/CommonDomain_webtier/bin` and enter the following:

```
WEBHOST1> ./opmnctl stopall
WEBHOST1> ./opmnctl startall
```

5. Repeat Steps 1 through 4 on *`WEBHOST2`*.

11.7 Validating the System

You should verify URLs to ensure that the appropriate routing and failover are working.

To verify the URLs:

1. Log in to the `FinancialDomain` Oracle WebLogic Server Administration Console and stop all the managed servers on the *`CRMHOST1`* while the managed servers on *`CRMHOST2`* are running.
2. Access the following URLs to verify that routing and failover are functioning properly. (Ensure the log in prompt is visible.)
   - `https://finexternal.mycompany.com/ledger/faces/JournalEntryPage`
   - `https://finexternal.mycompany.com/payables/faces/InvoiceWorkbench`
3. Log in to the `FinancialDomain` Oracle WebLogic Server Administration Console and stop all the managed servers on *`CRMHOST2`*.
4. Start the managed servers on *`CRMHOST1`*.
5. Repeat Step 2. (Ensure the log in prompt is visible.)
6. Start all the managed servers on *`CRMHOST2`* and verify that they are running on *`CRMHOST1`*.

12

Scaling Out the Oracle Fusion Incentive Compensation Domain

This chapter describes how to scale out the Oracle Fusion Incentive Compensation domain.

This chapter includes the following topics:

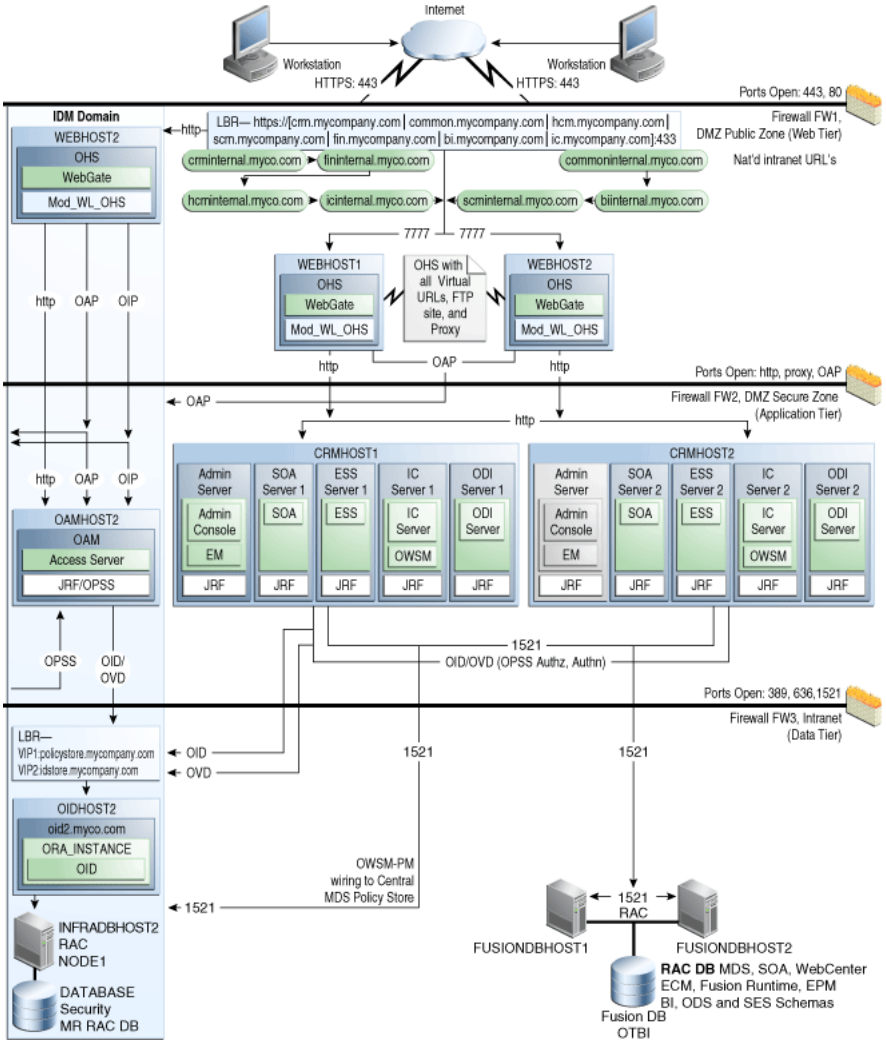
- Section 12.1, "Overview of the Oracle Fusion Incentive Compensation Domain"
- Section 12.2, "Prerequisites for Scaling Out the Oracle Fusion Incentive Compensation Domain"
- Section 12.3, "Adding a New Machine in the Oracle WebLogic Server Console"
- Section 12.4, "Packing and Unpacking the Managed Server Domain Home"
- Section 12.5, "Cloning Managed Servers and Assigning Them to CRMHOST2"
- Section 12.6, "Oracle HTTP Server Configuration"
- Section 12.7, "Validating the System"

12.1 Overview of the Oracle Fusion Incentive Compensation Domain

The Oracle Fusion Incentive Compensation domain provides Oracle Fusion Customer Relationship Management with a comprehensive solution that integrates with Quota Management to ensure consistent and accurate incentive compensation plan goals and enables organizations to measure and pay for performance that aligns with organizational strategies. Embedded participant and manager dashboards support sales performance management by enable salespeople to easily track their performance achievements against objectives and allowing them to focus on customers instead of shadow accounting.

Figure 12–1 shows the Oracle Fusion Incentive Compensation domain within the overall reference enterprise deployment topology for Oracle Fusion Applications.

Figure 12–1 Reference Topology for Oracle Fusion Incentive Compensation



12.2 Prerequisites for Scaling Out the Oracle Fusion Incentive Compensation Domain

Before you begin, ensure the following:

- Node Manager has been started in the Secure Sockets Layer (SSL) mode by following the instructions in Chapter 6, "Configuring Node Manager"
- You are starting with a clean machine if it is the first time it is being used for a scale out
- The `/etc/hosts` file has proper entries. To verify, ping this machine with the fully qualified name of the machine
- The user created on *CRMHOST2* should the same as the user on *CRMHOST1*
- The directory structure `/u01/oracle` is mounted to same shared file system as *CRMHOST1*
- The directory structure `/u02/local/oracle/config` on *CRMHOST2* has been created

- The initial Oracle Fusion Customer Relationship Management deployment on *CRMHOST1* has already been done and verified by provisioning

12.3 Adding a New Machine in the Oracle WebLogic Server Console

To add a new machine:

1. Log in to the Administration Server: `http://icinternal.mycompany.com:7777/console`.
2. Navigate to **ICDomain > Environment > Machines**.

   LocalMachine is located in the right-hand pane.
3. In the left-hand pane, click **Lock & Edit**.
4. In the right-hand pane, first click **New** to add the remote machine, and then specify the following:
   - Name - enter *CRMHOST2*
   - Machine operating system - Unix
5. Click **Next**.
6. In the window that opens, set the following attributes:
   - Type - SSL
   - Listen Address - *<CRMHOST2>*

     Note: The "localhost" default value here is wrong.
   - Listen port - 5556
7. Click **Finish** and activate the changes.

Note: If you get an error when activating the changes, see Section 18.8.18, "Administration Console Redirects from Internal URL to Container URL after Activation" for the temporary solution.

12.4 Packing and Unpacking the Managed Server Domain Home

Since the *CRMHOST1* domain directory file system is also available from *CRMHOST2*, both the `pack` and `unpack` commands can be executed from the *CRMHOST2*.

1. Do the following:
   a. Change directory to *ORACLE_BASE*`/products/fusionapps/oracle_common/common/bin`.
   b. Run the `pack` command:

```
CRMHOST2> ./pack.sh -managed=true -domain=ORACLE_BASE/config/domains/
CRMHOST1/ICDomain -template=ORACLE_BASE/user_templates/ICDomain_managed.jar
```

```
-template_name="IC_Managed_Server_Domain"
```

2. Ensure that `/u02/local/oracle/config/domains/`*`CRMHOST2`*`/ICDomain` is empty, and then run the `unpack` command:

   ```
   CRMHOST2> ./unpack.sh -domain=/u02/local/oracle/config/domains/
   CRMHOST2/ICDomain -template=ORACLE_BASE/user_templates/ICDomain_managed.jar
   ```

   Here, *`ORACLE_BASE`* is shared, and `/u02/local` is local to *`CRMHOST2`*.

12.5 Cloning Managed Servers and Assigning Them to CRMHOST2

To add a managed server and assign it to *`CRMHOST2`*:

1. Log in to the Administration Server: `http://`*`icinternal`*`.mycompany.com:7777/console`.
2. Navigate to **ICDomain > Environment > Servers**.
3. Switch to **Lock & Edit** mode.
4. Select the ***Managed_Servers*** checkbox (for example, **IncentiveCompensationServer_1**) and then click **Clone**.
5. Specify the following Server Identity attributes:
   - Server Name - `IncentiveCompensationServer_2`

     Note: To ensure consistency in naming, copy the name of the server shown in **Server Identity** and paste it into the **Server Name** field. Then change the number to "_2".

   - Server Listen Address - <*`CRMHOST2`*>
   - Server Listen Port - leave "as is"
6. Click **OK**.

   You now should see the newly cloned server, `IncentiveCompensationServer_2`.
7. Click **IncentiveCompensationServer_2** and change the following attributes:
   - Machine - <*`CRMHOST2`*>
   - Cluster Name - Default, IncentiveCompensationCluster
8. Click **Save** and then **Activate Changes**.
9. From the **Name** column, click the **IncentiveCompensationServer_2** scaled-out server link.
10. Click **Lock & Edit**, and then choose the **Keystores** tab.
11. Ensure that the keystores value is **Custom Identity and Custom Trust**.
12. Do the following:
    a. Change the Custom Identity Keystore path to point to the *`ORACLE_BASE`*`/products/fusionapps/wlserver_10.3/server/lib/`*`CRMHOST2`*`_fusion_identity.jks` file.
    b. Leave the Custom Identity Keystore type blank.

c. Change the Custom Identity Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

d. Re-enter the Confirm Custom Identity Keystore Passphrase.

e. Ensure that the Confirm Custom Trust Keystore path is pointing to the *ORACLE_BASE*/products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks file.

f. Leave the Custom Trust Keystore type blank.

g. Change the Custom Trust Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

h. Re-enter the Custom Trust Keystore Passphrase.

i. Click **Save**.

13. Choose the **SSL** tab.

a. Make sure that Identity and Trust Locations is set to **Keystores**.

b. Change the Private Key Alias to *CRMHOST2*_fusion.

c. Change the Private Key Passphrase to the *keypassword*, as described in the second bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

d. Re-enter the *keypassword* from Step c for the Confirm Private Key Passphrase.

e. Click **Save**.

14. Click **Activate Changes**.

15. Repeat Steps 2 to 14 for all the managed servers on this domain.

16. Set the following environment variable:

```
WLST_PROPERTIES="-Dweblogic.security.SSL.trustedCAKeyStore=ORACLE_BASE/
products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks"
```

17. Stop the domain's Administration Server:

```
CRMHOST1> ORACLE_BASE/config/domains/CRMHOST1/ICDomain/bin/stopWebLogic.sh
```

18. Restart the domain's Administration Server:

```
CRMHOST2> ORACLE_BASE/products/fusionapps/wlserver_10.3/common/bin/wlst.sh

CRMHOST2> nmConnect(username='<username>', password='<password>',
domainName='ICDomain', host='CRMHOST1',port='5556',
nmType='ssl', domainDir='ORACLE_BASE/config/domains/CRMHOST1/ICDomain')

CRMHOST2> nmStart('AdminServer')
```

> **Note:** The *username* and *password* used in the `nmConnect` are the Node Manager credentials (username and password) specified when creating the provisioning plan. This is shown in Figure 4–3 in "Using the Provisioning Process to Install Components for an Enterprise Deployment".

19. Run the newly created managed server:

a. Navigate to **ICDomain > Environment > Servers > Control**.

b. Select the newly created managed server and click **Start**.

c. Navigate to **ICDomain > Environment > Servers** and check the **State** to verify that the newly created managed servers are running.

20. Log in to the Administration Server once again (`http://icinternal.mycompany.com:7777/console`) and verify that all the managed servers, including scaled-out servers, are running.

Note: For all the scaled-up and scaled-out servers, change the Arguments in the `/u02/local/oracle/config/domains/HOSTNAME/DomainName/servers/ManagedServer/data/nodemanager/startup.properties` file to the following:

```
Arguments=-DJDBCProgramName\=DS/ICDomain/
IncentiveCompensationServer_2 -Dserver.group\
=IncentiveCompensationCluster
```

Note: For all the scaled-up and scaled-out managed servers, do the following:

1. Access the Oracle WebLogic Server Administration Console for the ICDomain:

   `http://icinternal.mycompany.com:7777/console`

2. Navigate to **Environment > Servers** and click the "Managed Server" link.
3. First select the **Logging** tab and then the **HTTP** tab.
4. Update the following parameters:

   – Log file name: logs/access.log.%yyyyMMdd%

   – Rotation Type: By Time

   – Limit number of retained files: leave this option unchecked

   – Rotate log file on startup: leave this option unchecked

5. Click **Save**.
6. Expand **Advanced Node** and set the following:

   – Format: Extended

   – Extended Logging Format Fields:

   ```
   date time time-taken cs-method cs-uri sc-status
   sc(X-ORACLE-DMS-ECID) cs(ECID-Context) cs(Proxy-Remote-User)
   cs(Proxy-Client-IP)
   ```

7. Click **Save** and **Activate Changes**.
8. Restart the Managed Server for the changes to take affect.

12.6 Oracle HTTP Server Configuration

To configure Oracle HTTP Server:

1. On *WEBHOST1*:

a. Change directory to *ORACLE_BASE*/config/CommonDomain_webtier/config/OHS/ohs1/moduleconf.

b. Copy FusionVirtualHost_ic.conf to FusionVirtualHost_ic.conf.org.

2. Edit the FusionVirtualHost_ic.conf file, adding the scaled-out host and port to all the WebLogic Application Clusters. Example 12–1 shows sample code for IncentiveCompensationServer.

Example 12–1 Sample "IncentiveCompensationServer" Code

```
<Location /IncentiveCompensation >
    SetHandler weblogic-handler
    WebLogicCluster <CRMHOST1:port>,<CRMHOST2:port>
</Location>
```

3. Repeat Step 2 for all applications.

4. Restart Oracle HTTP Server: cd to *ORACLE_BASE*/config/CommonDomain_webtier/bin and enter the following:

```
WEBHOST1> ./opmnctl stopall
WEBHOST1> ./opmnctl startall
```

5. Repeat Steps 1 through 4 on *WEBHOST2*.

12.7 Validating the System

You should verify URLs to ensure that the appropriate routing and failover are working.

To verify the URLs:

1. Log in to the ICDomain Oracle WebLogic Server Administration Console and stop all the managed servers on the *CRMHOST1* while the managed servers on *CRMHOST2* are running.

2. Access the following URL to verify that routing and failover are functioning properly. (Ensure the log in prompt is visible.)

   - https://icexternal.mycompany.com/incentiveCompensation/faces/IcCnCompPlanWorkarea

3. Log in to the ICDomain Oracle WebLogic Server Administration Console and stop all the managed servers on *CRMHOST2*.

4. Start the managed servers on *CRMHOST1*.

5. Repeat Step 2. (Ensure the log in prompt is visible.)

6. Start all the managed servers on *CRMHOST2* and verify that they are running on *CRMHOST1*.

13

Scaling Out the Oracle Business Intelligence Domain

This chapter describes how to scale out the Oracle Business Intelligence domain.

This chapter includes the following topics:

- Section 13.1, "Overview of the Oracle Business Intelligence Domain"
- Section 13.2, "Prerequisites for Scaling Out the Oracle Business Intelligence Domain"
- Section 13.3, "Starting the Default Node Manager"
- Section 13.4, "Prerequisites for Scaling Out Oracle Business Intelligence on CRMHOST2"
- Section 13.5, "Scaling Out Oracle Business Intelligence Components"
- Section 13.6, "Configuring Oracle Essbase Clustering Using the Essbase Failover Automation Tool"
- Section 13.7, "Validating the System"

13.1 Overview of the Oracle Business Intelligence Domain

Oracle Fusion Customer Relationship Management Sales and Marketing offerings use following Oracle Business Intelligence components from the Oracle Business Intelligence domain:

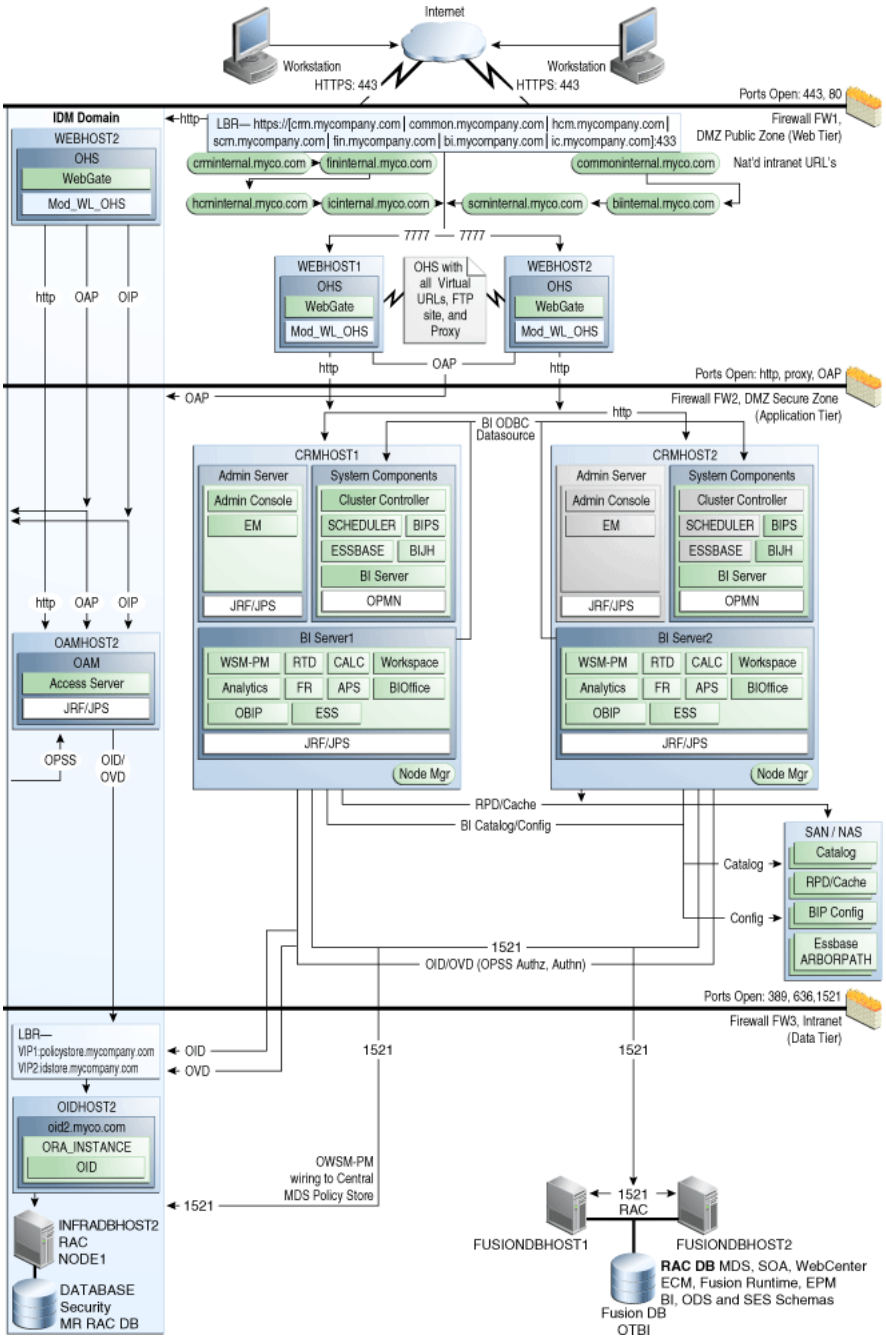
- Oracle Business Intelligence Analytics
- Essbase
- Oracle Real-Time Decisions
- Oracle Business Intelligence Publisher

Key components of Oracle Fusion Customer Relationship Management, such as Territory (which is the core component of Oracle Fusion Customer Relationship Management), rely on Oracle Business Intelligence Analytics as the source data for planning and optimization.

Oracle Business Intelligence Analytics is the mandatory underlying component for Territory Management, Sales Predictor Engine, and Opportunity Landscape, as they are analytic-centric applications.

Figure 13–1 shows the Oracle Business Intelligence domain within the overall reference enterprise deployment topology for Oracle Fusion Applications.

Figure 13–1 Reference Topology for Oracle Business Intelligence Domain



13.2 Prerequisites for Scaling Out the Oracle Business Intelligence Domain

Before you begin, ensure the following:

- Node Manager has been started in the Secure Sockets Layer (SSL) mode by following the instructions in Chapter 6, "Configuring Node Manager"
- You are starting with a clean machine if it is the first time it is being used for a scale out

- The `/etc/hosts` file has proper entries. To verify, ping this machine with the fully qualified name of the machine
- The user created on *CRMHOST2* should the same as the user on *CRMHOST1*
- The directory structure `/u01/oracle` is mounted to same shared file system as *CRMHOST1*
- The directory structure `/u02/local/oracle/config` on *CRMHOST2* has been created
- The initial Oracle Fusion Customer Relationship Management deployment on *CRMHOST1* has already been done and verified by provisioning
- The Administration Console's **Follow Configuration Changes** feature has been disabled (to eliminate redirections):
  1. Log into the Administration Console and go to **Preferences > Shared Preferences**.
  2. Deselect **Follow Configuration Changes** and click **Save**.

13.3 Starting the Default Node Manager

To start the default Node Manager:

1. Stop any Node Manager running on *CRMHOST2*.
2. Change directory to *ORACLE_BASE*`/products/fusionapps/wlserver_10.3/common/nodemanager` and edit the `nodemanager.properties` file with the following:

   ```
   SecureListener=false
   ```

3. Change directory to *ORACLE_BASE*`/products/fusionapps/oracle_common/common/bin` and run the following script:

   ```
   ./setNMProps.sh
   ```

4. Change directory to *ORACLE_BASE*`/products/fusionapps/wlserver_10.3/server/bin` and run the following script:

   ```
   ./startNodeManager.sh
   ```

   Node Manager starts on *CRMHOST2*.

> **Note:** Steps 2 through 4 will enable Node Manager on *CRMHOST2* and the Administrator Console to communicate on Plain Socket.

13.4 Prerequisites for Scaling Out Oracle Business Intelligence on CRMHOST2

Prerequisites include the following:

- Configuring JMS for Oracle BI Publisher
- Setting the Listen Address for bi_server1 Managed Server
- Updating the FusionVirtualHost_bi.conf Configuration File

13.4.1 Configuring JMS for Oracle BI Publisher

You must 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 in to the Administration Console.
2. In the Domain Structure window, expand the **Services** node and then 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 *CRMHOST1* and *CRMHOST2*:

   ORACLE_BASE/config/domains/*CRMHOST1*/BIDomain/BipJmsStore

5. Click **Save** and then click **Activate Changes**.

   The changes will not take effect until the Managed Server is restarted.
6. Do the following:
   a. Ensure that Node Manager is up and running.
   b. On the Summary of Servers page, select the **Control** tab.
   c. Select **bi_server1** in the table and then click **Shutdown**.
   d. After the server has shut down, select **bi_server1** in the table and then click **Start**.
7. Run the following commands to restart the Oracle Business Intelligence system components:

```
$ cd /u02/local/oracle/config/BIInstance/bin

$ ./opmnctl restartproc
```

13.4.2 Setting the Listen Address for bi_server1 Managed Server

Make sure that you have performed the steps described in Section 14.1, "Enabling Virtual IPs on CRMHOST1 and CRMHOST2" before setting the `bi_server1` listen address.

To set the listen address for the Managed Server:

1. Log in to the Administration Console (`http://biinternal.mycompany.com:7777/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 column of the table. The Setting page for `bi_server1` is displayed.
6. Set the **Listen Address** to *BIVH1*.
7. Click **Save**.
8. Click **Activate Changes**.

9. The changes will not take effect until the `bi_server1` Managed Server is restarted (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. 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 /u02/local/oracle/config/BIInstance/bin
$ ./opmnctl stopall
$ ./opmnctl startll
```

13.4.3 Updating the FusionVirtualHost_bi.conf Configuration File

To enable Oracle HTTP Server to route to `bi_cluster`, which contains the `bi_servern` Managed Servers, you must set the WebLogicCluster parameter to the list of nodes in the cluster:

1. On *WEBHOST1* and *WEBHOST2*, update the WebLogicCluster parameter in the *ORACLE_BASE*`/config/CommonDomain_webtier`*n*`/config/OHS/ohs1/moduleconf/FusionVirtualHost_bi.conf` file to contain a cluster list of Virtual host:port entries.

   For example,

```
<LocationMatch ^/analytics/>
SetHandler weblogic-handler
WebLogicCluster BIVH1:10217,BIVH2:10217
</LocationMatch>
```

2. On *WEBHOST1* and *WEBOST2*, add the following context roots to the *ORACLE_BASE*`/config/CommonDomain_webtier`*n*`/config/OHS/ohs1/moduleconf/FusionVirtualHost_bi.conf` file:

```
RedirectMatch 301 ^/biofficeclient$ /biofficeclient/

RedirectMatch 301 ^/bimiddleware$ /bimiddleware/

RedirectMatch 301 ^/rtis$ /rtis/

RedirectMatch 301 ^/bisearch$ /bisearch/

<LocationMatch ^/biofficeclient/>
SetHandler weblogic-handler
WebLogicCluster BIVH1:10217,BIVH2:10217
</LocationMatch>

<LocationMatch ^/bimiddleware/>
SetHandler weblogic-handler
WebLogicCluster BIVH1:10217,BIVH2:10217
</LocationMatch>

<LocationMatch ^/rtis/>
SetHandler weblogic-handler
WebLogicCluster BIVH1:10217,BIVH2:10217
</LocationMatch>
```

```
<LocationMatch ^/bisearch/>
SetHandler weblogic-handler
WebLogicCluster BIVH1:10217,BIVH2:10217
</LocationMatch>
```

3. Restart Oracle HTTP Server on both *WEBHOST1* and *WEBHOST2*, as follows:

```
WEBHOST1> ORACLE_BASE/config/CommonDomain_webtier/bin/opmnctl restartproc
ias-component=ohs1
WEBHOST2> ORACLE_BASE/config/CommonDomain_webtier1/bin/opmnctl restartproc
ias-component=ohs1
```

The servers specified in the WebLogicCluster parameters are only important 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 then add a third member, you do not need to update the configuration to add the third member. The third member will be discovered dynamically at run time.
- **Example 2:** You have a three-node cluster, but only two nodes are listed in the configuration. However, if both listed nodes are down when you start Oracle HTTP Server, then the plug-in would fail to route to the cluster. You must ensure that at least one of the listed nodes is running when you start Oracle HTTP Server.

If you list all the members of the cluster, then you guarantee you can route to the cluster, assuming at least one member is running when Oracle HTTP Server is started. For more information on configuring the WebLogic Server plug-in, see *Oracle Fusion Middleware Using Web Server Plug-Ins with Oracle WebLogic Server*.

13.5 Scaling Out Oracle Business Intelligence Components

This section describes how to scale out the Oracle Business Intelligence system using the Configuration Assistant. It is assumed that an Oracle Business Intelligence *ORACLE_BASE* (binaries) has already been installed and is available from *CRMHOST1* and *CRMHOST2*, and that a domain with an Administration Server has been created. This is the domain that will be extended in this chapter to support Oracle Business Intelligence components.

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

This section includes the following topics:

- Section 13.5.1, "Scaling Out the BI System on CRMHOST2"
- Section 13.5.3, "Scaling Out the System Components"
- Section 13.5.4, "Configuring Secondary Instances of Singleton System Components"

- Section 13.5.5, "Configuring the bi_server2 Managed Server"
- Section 13.5.6, "Performing Additional Configuration for Oracle Business Intelligence High Availability"
- Section 13.5.7, "Configuring a Default Persistence Store for Transaction Recovery"
- Section 13.5.8, "Starting and Validating Oracle Business Intelligence on CRMHOST2"
- Section 13.5.9, "Validating Access Through Oracle HTTP Server"
- Section 13.5.10, "Configuring Node Manager for the Managed Servers"
- Section 13.5.11, "Configuring Server Migration for the Managed Servers"

13.5.1 Scaling Out the BI System on CRMHOST2

Do the following to scale out the Oracle Business Intelligence system:

1. Change directory to the location of the Configuration Assistant:

   ```
   CRMHOST2> ORACLE_BASE/products/fusionapps/bi/bin
   ```

2. Start the Oracle Business Intelligence Configuration Assistant:

   ```
   CRMHOST2> ./config.sh
   ```

3. In the Welcome screen, click **Next**.
4. In the Prerequisite Checks screen, verify that all checks complete successfully, and then click **Next**.
5. In the Create or Scale out BI System screen, select **Scale Out BI System** and enter the following:
   - **Host Name:** *CRMHOST1*
   - **Port:** 10201
   - **User name:** *WLS_Administrator*
   - **User Password:** *WLS_Administrator_password*

   Click **Next**.
6. In the Scale Out BI System Details screen, enter the following:
   - **Middleware Home:** *ORACLE_BASE* /products/fusionapps (dimmed)
   - **Oracle Home:** *ORACLE_BASE*/products/fusionapps/bi (dimmed)
   - **WebLogic Server Home:** *ORACLE_BASE*/products/fusionapps/wlserver_10.3 (dimmed)
   - **Domain Home:** /u02/local/oracle/config/domains/*CRMHOST1* /BIDomain
   - **Applications Home:** /u02/local/oracle/config/applications/*CRMHOST1*/BIDomain
   - **Instance Home:** Defaults to /u02/local/oracle/config/BIInstance1
   - **Instance Name:** BIInstance1 (dimmed)

   Click **Next**.

7. In the Configure Ports screen, select "Specify Ports using Configuration File."

   Use the `bi_staticports.ini` file from the *ORACLE_BASE*`/products/ports` directory.

   Click **Next**.

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

   Click **Next**.

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

10. In the Configuration Progress screen, verify that all the Configuration Tools have completed successfully and click **Next**.

11. In the Complete screen, click **Finish**.

13.5.2 Start the Node Manager in SSL Mode

Do the following:

1. Shutdown the default node manager.

2. Start the Node Manager in SSL mode on *CRMHOST2*:

```
CRMHOST2> cd ORACLE_BASE/products/fusionapps/wlserver_10.3/common/nodemanager/
          CRMHOST2

CRMHOST2> ./startNodeManagerWrapper.sh
```

3. Update the Node Manager for the *CRMHOST2* machine using the Oracle WebLogic Server Console by doing the following:

   a. Log in to the Administration Server: `http://biinternal.mycompany.com:7777/console`.

   b. Navigate to **BIDomain> Environment > Machines**.

   c. In the left-hand pane, click **Lock & Edit**.

   d. In the right-hand pane, click *CRMHOST2*.

   e. In the window that opens, click the **Node Manager** tab and set the following attributes:

      – Type - SSL

      – Listen Address - <*CRMHOST2*>

      – Listen Port - 5556

4. Click **Save** and then **Activate Changes**.

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

5. Do the following to restart the `bi_server2` Managed Server:

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

   b. Select **bi_server2** in the table and then click **Shutdown**.

   c. After the server has shut down, select **bi_server2** in the table and then click **Start**.

13.5.3 Scaling Out the System Components

Do the following in Oracle Enterprise Manager Fusion Middleware Control:

1. Log in to Fusion Middleware Control (`http://biinternal.mycompany.com:7777/em`).
2. Expand the **Business Intelligence** node in the Farm_BIDomain window.
3. Click **coreapplication**.
4. Click **Capacity Management**, then click **Scalability**.
5. Click **Lock and Edit Configuration**.
6. For the *CRMHOST2* BIInstance1 Oracle instance, increment the Oracle Business Intelligence components by 1:
   - BI Servers
   - Presentation Servers
   - JavaHosts
7. Change the **Port Range From** and **Port Range To** to be the same as the *CRMHOST1* BIInstance Oracle instance.

   Port Range To on both instances must be increased from 10214 to 10215.

8. Click **Apply**.
9. Click **Activate Changes**.

You do not need to restart at this point, because you will perform a restart after completing the steps in Section 13.5.4, "Configuring Secondary Instances of Singleton System Components."

13.5.4 Configuring Secondary Instances of Singleton System Components

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

In Fusion Middleware Control:

1. Log in to Fusion Middleware Control (`http://biinternal.mycompany.com:7777/em`).
2. Expand the **Business Intelligence** node in the Farm_BIDomain window.
3. Click **coreapplication**.
4. Click **Capacity Management**, then click **Availability**.
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 and BI Cluster Controller.
7. Click **Apply**.

   Under Potential Single Points of Failure, it should report **No problems - all components have a backup**.

8. Click **Activate Changes**.

9. Click **Restart to apply recent changes**.
10. From **Manage System**, click **Restart**.
11. Click **Yes** when prompted to confirm that you want to restart all Business Intelligence components.

13.5.5 Configuring the bi_server2 Managed Server

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

- Section 13.5.5.1, "Setting the Listen Address for the bi_server2 Managed Server"
- Section 13.5.5.2, "Configuring Custom Identity and Custom Trust for the bi_server2 Managed Server"
- Section 13.5.5.3, "Disabling Host Name Verification for the bi_server2 Managed Server"

13.5.5.1 Setting the Listen Address for the bi_server2 Managed Server

Make sure that you have performed the steps described in Section 14.1, "Enabling Virtual IPs on CRMHOST1 and CRMHOST2" before setting the `bi_server2` listen address.

To set the listen address for the Managed Server:

1. Log in to the Oracle WebLogic Server Administration Console (`http://biinternal.mycompany.com:7777/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 column of the table. The settings page for `bi_server2` is displayed.
6. Set the **Listen Address** to *`BIVH2`*.
7. Click **Save**.
8. Click **Activate Changes**.

   The changes will not take effect until the Managed Server is restarted.
9. Do the following:
   a. Ensure that Node Manager is up and running.
   b. On the Summary of Servers page, select the **Control** tab.
   c. Select **bi_server2** in the table and then click **Shutdown**.
   d. After the server has shut down, select **bi_server2** in the table and then click **Start**.

13.5.5.2 Configuring Custom Identity and Custom Trust for the bi_server2 Managed Server

Do the following:

1. Log in to the Oracle WebLogic Server Administration Console (`http://biinternal.mycompany.com:7777/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 displays.
5. Select **bi_server2** in the column of the table. The Settings page for `bi_server2` displays.
6. Click **Keystores**, and then do the following:
   a. Click **Change** next to **Demo Identity and Demo Trust**.
   b. Select **Custom Identity and Custom Trust** from the **Keystores** dropdown list and click **Save**.
   c. Under **Identity**, do the following:

      – Change the Custom Identity Keystore entry to point to the *ORACLE_BASE*`/products/fusionapps/wlserver_10.3/server/lib/CRMHOST2_fusion_identity.jks` file.

      – Enter and confirm the Custom Identity Keystore Passphrase. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."
   d. Under **Trust**, do the following:

      – Change the Custom Identity Keystore entry to point to the *ORACLE_BASE*`/products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks` file.

      – Enter and confirm the Custom Trust Keystore Passphrase. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

      – Click **Save**.
7. Click **SSL**, and then do the following:
   a. Ensure that Identity and Trust Locations is set to **Keystores**.
   b. Under **Identity**, do the following:

      – Change the Private Key Alias to *`CRMHOST2_fusion`*.

      – Enter and confirm the Private Key Passphrase to the *keypassword*, as described in the second bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

      – Click **Save**.
8. Click **Activate Changes**.
9. Set the following property in *ORACLE_BASE*`/products/fusionapps/wlserver_10.3/common/bin/wlst.sh`:

```
WLST_PROPERTIES=" -Dweblogic.wlstHome='${WLST_HOME}'
-Dweblogic.security.SSL.trustedCAKeyStore=ORACLE_BASE/
products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks ${WLST_
PROPERTIES}"
```

13.5.5.3 Disabling Host Name Verification for the bi_server2 Managed Server

This step is required if you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server (see Chapter 6, "Configuring Node Manager"). If you have not configured the server certificates, you will receive errors when managing the different WebLogic servers. To avoid these errors, disable host name verification while setting up and validating the topology, and enable it again once the enterprise deployment topology configuration is complete as described in Chapter 6, "Configuring Node Manager."

To disable host name verification:

1. Log in to Oracle WebLogic Server Administration Console (`http://biinternal.mycompany.com:7777/console`).
2. 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 column of the table. The settings page for the server is displayed.
6. Click 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 will not take effect until the `bi_server2` Managed Server is restarted (make sure 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. Restart **bi_server2**.
12. Restart the BI System Components:

```
$ cd /u02/local/oracle/config/BIInstance1/bin
$ ./opmnctl stopall
$ ./opmnctl startall
```

13.5.6 Performing Additional Configuration for Oracle Business Intelligence High Availability

This section describes additional high availability configuration tasks for Oracle BI Enterprise Edition, Oracle Real-Time Decisions, Oracle BI Publisher, and Oracle Financial Reports. It includes the following topics:

- Section 13.5.6.1, "Additional Configuration Tasks for Oracle BI Scheduler"
- Section 13.5.6.2, "Additional Configuration Tasks for Oracle Real-Time Decisions"
- Section 13.5.6.3, "Additional Configuration Tasks for Oracle BI Publisher"
- Section 13.5.6.4, "Additional Configuration Tasks for Oracle BI for Microsoft Office"

- Section 13.5.6.5, "Additional Configuration Tasks for Oracle Financial Reporting"

13.5.6.1 Additional Configuration Tasks for Oracle BI Scheduler

If you use server-side scripts with Oracle BI Scheduler, 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 these steps only if you are using server-side scripts.

To share Oracle BI Scheduler scripts:

1. Create an *ORACLE_BASE*/config/BIShared/OracleBISchedulerComponent/coreapplication_obisch1 directory .
2. Copy the default Oracle BI Scheduler scripts (for example, /u02/local/oracle/config/BIInstance/bifoundation/OracleBISchedulerComponent/coreapplication_obisch1/scripts/common) and custom Oracle BI Scheduler scripts (for example, /u02/local/oracle/config/BIInstance/bifoundation/OracleBISchedulerComponent/coreapplication_obisch1/scripts/scheduler) to the following location:

   ORACLE_BASE/config/BIShared/OracleBISchedulerComponent/coreapplication_obisch1
3. 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.

   The instanceconfig.xml files for Oracle BI Scheduler are in the following locations:

   On *CRMHOST1*: /u02/local/oracle/config/BIInstance/config/OracleBISchedulerComponent/coreapplication_obisch1

   On *CRMHOST2*: /u02/local/oracle/config/BIInstance1/config/OracleBISchedulerComponent/coreapplication_obisch1

   You must update these files for each Oracle BI Scheduler component in the deployment.
4. Restart the Oracle BI Scheduler component.

   On *CRMHOST1*:

   ```
   $ cd /u02/local/oracle/config/BIInstance/bin
   $ ./opmnctl stopproc
   ias-component=coreapplication_obisch1
   $ ./opmnctl startproc
   ias-component=coreapplication_obisch1
   ```

   On *CRMHOST2*:

   ```
   $ cd /u02/local/oracle/config/BIInstance1/bin
   $ ./opmnctl stopproc
   ```

```
ias-component=coreapplication_obisch1
$ ./opmnctl startproc
ias-component=coreapplication_obisch1
```

13.5.6.2 Additional Configuration Tasks for Oracle Real-Time Decisions

This sections contains the following topics:

- Section 13.5.6.2.1, "Configuring Oracle Real-Time Decisions Clustering Properties"
- Section 13.5.6.2.2, "Adding System Properties to the Server Start Tab"

13.5.6.2.1 Configuring Oracle Real-Time Decisions Clustering Properties Perform these steps in Fusion Middleware Control to set up cluster-specific configuration properties for Oracle RTD. You only need to perform the steps on one of the nodes in your deployment. You do not need to set cluster-specific configuration properties for Oracle RTD for subsequent nodes.

1. Log in to Fusion Middleware Control (`http://biinternal.mycompany.com:7777/em`).
2. Expand the **Application Deployments** node in the Farm_BIDomain 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 MBean and set the attribute, as shown in Table 13–1. Other servers automatically get updated with the value you set.

Table 13–1 Oracle RTD MBean Attributes and Values for Clustering

MBean	Attribute	Value
SDClusterPropertyManager -> Misc	DecisionServiceAddress	http://biinternal.mycompany.com:7777

8. Click **Apply**.

13.5.6.2.2 Adding System Properties to the Server Start Tab After scaling out Oracle RTD, use the Administration Console to add three system properties to the **Server Start** tab of each Managed Server.

In the Administration Console, choose **Environment > Servers > bi_server<1,2> > Server Start > Arguments** and then add these three properties:

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

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, the Oracle RTD batch framework will be in an inconsistent state. If this is the case, do the following:

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 **BatchManager** MBean, under either **Application Defined MBeans > OracleRTD > Server:bi_server1 > Server** or under **Application Defined MBeans > OracleRTD > Server:bi_server2> Server**.

   It should be in one of these locations, but not both.
3. Locate the corresponding MBean attribute **BatchManagerEnabled** under **SDPropertyManager > Misc**. For example, if the BatchManager MBean is located in `bi_server1`, then select **Application Defined MBeans > OracleRTD > Server:bi_server1 > SDPropertyManager > Misc : BatchManagerEnabled**.
4. Set the BatchManagerEnabled attribute to **false** and click **Apply**.
5. Set the BatchManagerEnabled attribute back to **true** and click **Apply**. Performing this task causes the Batch Manager to stop and be restarted.

   When it restarts, it will be running on either the same server as before, or on a different server.
6. After restarting Batch Manager, note that the corresponding MBean does not always immediately get refreshed on the server where Batch Manager comes back up, so this is not a concern. Instead, verify that Batch Manager is now operational by using the Batch Console tool:

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

      ORACLE_BASE/products/fusionapps/bi/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 machine (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 preceeding example.

13.5.6.3 Additional Configuration Tasks for Oracle BI Publisher

Perform the steps in this section on each machine where Oracle BI Publisher is configured.

This section includes the following topics:

- Section 13.5.6.3.1, "Configuring Integration with Oracle BI Presentation Services"
- Section 13.5.6.3.2, "Setting the Oracle BI EE Data Source"
- Section 13.5.6.3.3, "Configuring JMS for BI Publisher"

13.5.6.3.1 Configuring Integration with Oracle BI Presentation Services To configure Oracle BI Publisher integration with Oracle BI Presentation Services:

1. Log in to Oracle BI Publisher (`http://biinternal.mycompany.com:7777/xmlpserver`) 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:** 7777
   - **URL Suffix:** analytics-ws/saw.dll
4. Click **Apply**.
5. Restart your Oracle BI Publisher application:
   a. Log in to the Administration Console (`http://biinternal.mycompany.com:7777/console`).
   b. Click **Deployments** in the Domain Structure window.
   c. Select **bipublisher(11.1.1)**.
   d. Click **Stop**.
   e. After the application has stopped, click **Start**.

13.5.6.3.2 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 Oracle BI Publisher.

To set the Oracle BI EE data source in Oracle BI Publisher:

1. Log in to Oracle BI Publisher (`http://biinternal.mycompany.com:7777/xmlpserver`) 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://CRMHOST1:10212/PrimaryCCS=CRMHOST1;PrimaryCCSPort=10212;
   SecondaryCCS=CRMHOST2;SecondaryCCSPort=10212;
   ```

4. Deselect **Use System User** and specify administrator credentials for **Username** and **Password** (for example, FAadmin).

5. Click **Test Connection**. You should receive a "Connection established successfully" message.
6. Click **Apply**.

13.5.6.3.3 Configuring JMS for BI Publisher You must configure the location for all persistence stores to a directory visible from both nodes. Change all persistent stores to use this shared base directory.

On *CRMHOST2*:

1. Log in to the Oracle WebLogic Server Administration Console (`http://biinternal.mycompany.com:7777/console`).
2. In the Domain Structure window, expand the **Services** node and then click the **Persistence Stores** node. The Summary of Persistence Stores page is displayed.
3. Click **Lock & Edit**.
4. Click **New**, and then **Create File Store**.
5. Enter a name (for example, BipJmsStore2) and target (for example, BI _SERVER2). Enter a directory that is located in shared storage so that it is accessible from both *CRMHOST1* and *CRMHOST2*:

   ORACLE_BASE`/config/domains/`*CRMHOST1*`/BIDomain/BipJmsStore`
6. Click **OK** and then **Activate Changes**.
7. In the Domain Structure window, expand the **Services** node and then click the **Messaging > JMS Servers** node. The Summary of JMS Servers page is displayed.
8. 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 then click the **Messaging > JMS Modules** node. The JMS Modules page is displayed.
14. In the Change Center, click **Lock & Edit**.
15. Click **BIPJmsResource** and then click the **Subdeployments** tab.
16. Select **BipJmsSubDeployment** under **Subdeployments**.
17. Add the new Oracle BI Publisher JMS Server (**BipJmsServer2**) as an additional target for the subdeployment.
18. Click **Finish** and then **Activate Changes**.

13.5.6.4 Additional Configuration Tasks for Oracle BI for Microsoft Office

This section includes the following topics:

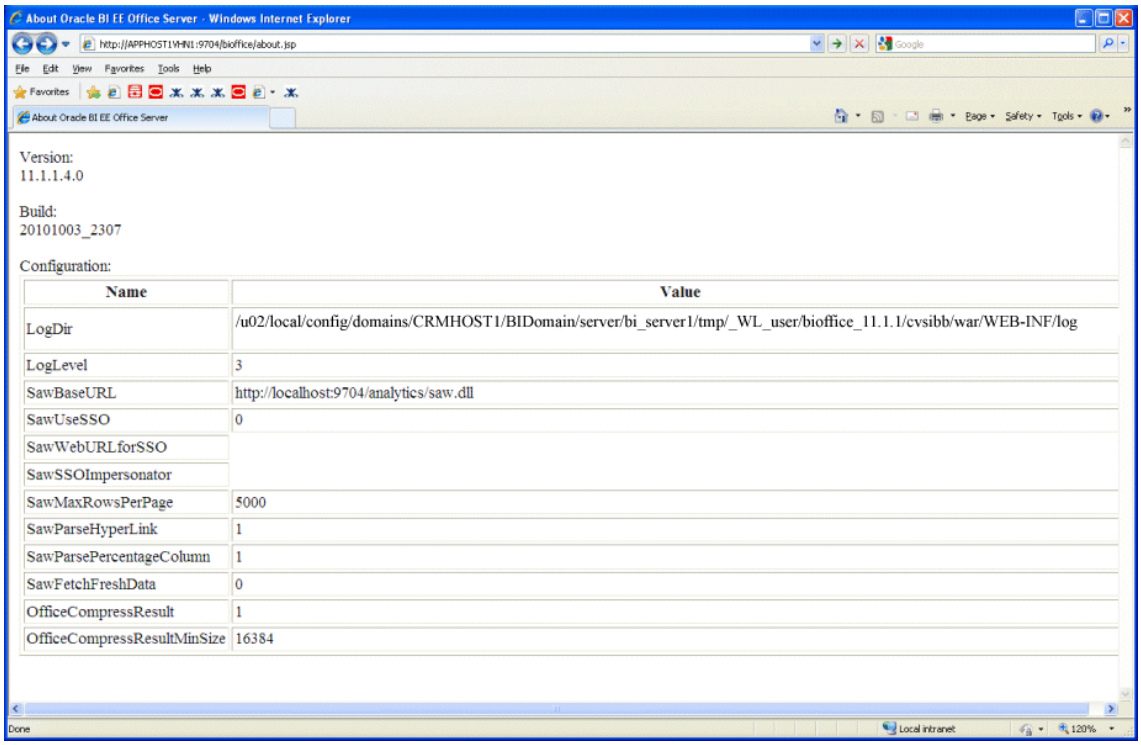
- Section 13.5.6.4.1, "Configuring Oracle BI for Microsoft Office Properties"
- Section 13.5.6.4.2, "Validating Oracle BI for Microsoft Office"

13.5.6.4.1 Configuring Oracle BI for Microsoft Office Properties To perform additional configuration tasks for Oracle BI for Microsoft Office:

1. Validate the Oracle BI Enterprise Edition Office Server setup by accessing `http://biinternal.mycompany.com:7777/bioffice/about.jsp`.

   The About Oracle BI EE Office Server page is displayed, as shown in Figure 13–2.

Figure 13–2 About Oracle BI EE Office Server Page



2. Go to the Oracle BI Enterprise Edition Office Server directory. For example:

   `/u02/local/config/domains/`*`CRMHOST1`*`/BIDomain/servers/bi_server1/tmp/_WL_user/bioffice_11.1.1/cvsibb/war/WEB-INF`

   If you are not sure how to locate the Oracle BI Enterprise Edition Office Server directory, check the **LogDir** parameter on the About Oracle BI EE Office Server page. The Oracle BI Enterprise Edition Office Server directory is the parent directory of the log directory.

3. On both *`CRMHOST1`* and *`CRMHOST2`*, open bioffice.xml for editing and modify the BI Office properties shown in Table 13–2.

Table 13–2 BI Office Properties in bioffice.xml

Property Name	Valid Value	Description
SawBaseURL	http://biinternal.mycompany.com:7777/analytics/saw.dll or http://biinternal.mycompany.com:7777/analytics-ws/saw.dll	Load Balancer Virtual Server Name URL for Oracle BI Presentation Services. **Important:** If SSO is enabled, then enter the URL for the protected analytics servlet that you deployed when configuring BI Office to integrate with the SSO-enabled Oracle BI Server. The URL that is specified for this property is used for Web services requests between the BI Office Server and Presentation Services.
SawUseSSO	0 = No (Default) 1 = Yes	Set this property to `1` if the Oracle Business Intelligence implementation is enabled for SSO.
SawWebURLforSSO	http://biinternal.mycompany.com:7777/analytics/saw.dll	When SSO is enabled, use this property to enter the public URL that allows external users to access Oracle Business Intelligence using SSO from the Oracle BI Add-in for Microsoft Office.

4. Restart the BI Office application:
   a. Log in to the Administration Console (`http://biinternal.mycompany.com:7777/console`).
   b. Click **Deployments** in the Domain Structure window.
   c. Select **bioffice(11.1.1)**.
   d. Click **Stop**.
   e. After the application has stopped, click **Start**.
5. Validate that the **SawBaseURL** parameter has been updated on the About Oracle BI EE Office Server page.

13.5.6.4.2 Validating Oracle BI for Microsoft Office To validate configuration for Oracle BI for Microsoft Office:

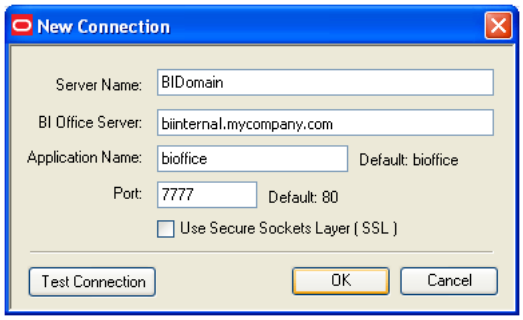
1. Log in to Oracle BI Presentation Services at:

   `http://biinternal.mycompany.com:7777/analytics`
2. In the lower left pane, under the Get Started heading, select **Download BI Desktop Tools** and then select **Oracle BI for MS Office**.
3. Install Oracle BI for Microsoft by running the Oracle BI Office InstallShield Wizard.
4. Open Microsoft Excel or Microsoft PowerPoint.
5. From the **Oracle BI** menu, select **Preferences**.
6. In the **Connections** tab, select **New**.
7. Enter values for the following fields:
   - **Server Name:** Provide a name for the connection.

- **BI Office Server:** Provide the URL for the Oracle BI Office Server.
- **Application Name:** Enter the Application Name that you defined for the Oracle BI Office Server when you deployed the Oracle BI Office Server application to WLS. The default name is **bioffice**.
- **Port:** Enter the Oracle BI Office Server port number.

Figure 13–3 shows the New Connection dialog.

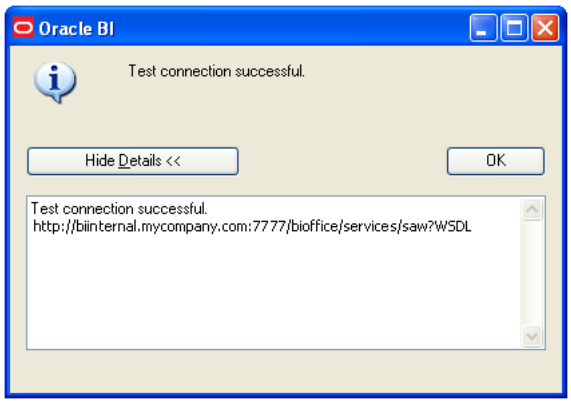
Figure 13–3 New Connection Dialog for Oracle BI Office



8. Click **Test Connection** to test the connection between the add-in and the Oracle BI Office Server.

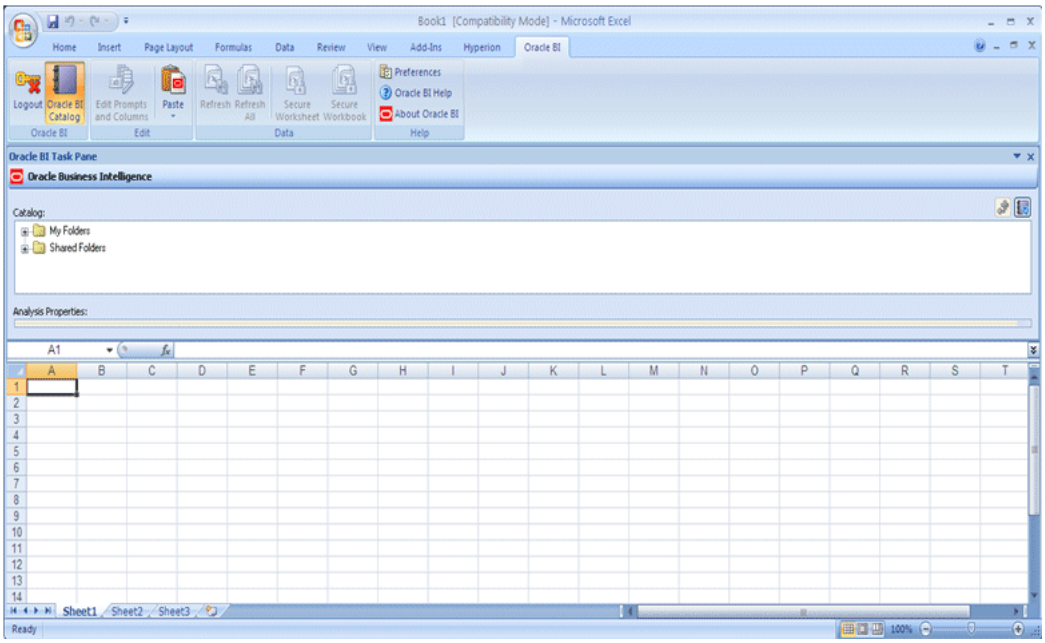
   Successful connections receive a "Test connection successful" message, as shown in Figure 13–4.

Figure 13–4 Test Connection Successful Message



9. Log in as an Administrator (for example, `weblogic`) and validate that you can access the Oracle BI Task Pane, as shown in Figure 13–5.

Figure 13–5 Oracle BI Task Pane in Microsoft Excel



13.5.6.5 Additional Configuration Tasks for Oracle Financial Reporting

There are additional configuration tasks to perform for Oracle Financial Reporting. Do the following on *CRMHOST1* and *CRMHOST2*:

1. Update the `VARIABLE_VALUE_LIMIT` from 30720 to 3072000 in the `NQSConfig.INI` file. For example,

   ```
   VARIABLE_VALUE_LIMIT = 3072000;
   ```

   On *CRMHOST1*, this file is located in `/u02/local/oracle/config/BIInstance/config/OracleBIServerComponent/coreapplication_obis1`.

   On *CRMHOST2*, this file is located in `/u02/local/oracle/config/BIInstance1/config/OracleBIServerComponent/coreapplication_obis1`.

2. Run the following commands to restart the Oracle Business Intelligence system components:

   ```
   $ cd /u02/local/oracle/config/BIInstancen/bin
   $ ./ompnctl stopall
   $ ./opmnctl startall
   ```

13.5.7 Configuring a Default Persistence Store for Transaction Recovery

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

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

To set the location for the default persistence store:

1. Log in to the Oracle WebLogic Server Administration Console (`http://biinternal.mycompany.com:7777/console`).
2. In the Change Center, click **Lock & Edit**.
3. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node. The Summary of Servers page is displayed.
4. Click the name of the server (represented as a hyperlink) in the 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 will store its data files. For example:

   CRMHOST1`> `*ORACLE_BASE*`/config/domains/`*CRMHOST1*`/BIDomain/tlogs`

7. Click **Save**.
8. Repeat Steps 1 through 7 for the `bi_server2` Managed Server.
9. Click **Activate Changes**.
10. Restart the Managed Servers to activate the changes (ensure that Node Manager is up and running):
    a. Log in to the Oracle WebLogic Server Administration Console (`http://biinternal.mycompany.com:7777/console`).
    b. In the Summary of Servers screen, select the **Control** tab.
    c. Select **bi_server1** and **bi_server2** in the table and then click **Shutdown**.
    d. Restart the `bi_server1` and `bi_server2` servers.
    e. Restart the Oracle Business Intelligence system components:

```
$ cd /u02/local/oracle/config/BIInstance/bin
$ ./ompnctl stopall
$ ./opmnctl startall
```

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.

13.5.8 Starting and Validating Oracle Business Intelligence on CRMHOST2

This section includes the following topics:

- Section 13.5.8.1, "Starting the bi_server2 Managed Server"
- Section 13.5.8.2, "Starting the Oracle Business Intelligence System Components"
- Section 13.5.8.3, "Validating Oracle Business Intelligence URLs"

13.5.8.1 Starting the bi_server2 Managed Server

To start the `bi_server2` Managed Server:

1. Start the `bi_server2` Managed Server using the Oracle WebLogic Server Administration Console, as follows:
   a. Log in to the Oracle WebLogic Server Administration Console (`http://biinternal.mycompany.com:7777/console`).
   b. Expand the **Environment** node in the **Domain Structure** window.
   c. Choose **Servers**. The Summary of Servers page is displayed.
   d. Click the **Control** tab.
   e. 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," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors.

13.5.8.2 Starting the Oracle Business Intelligence System Components

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

To start the Oracle Business Intelligence system components using the `opmnctl` command-line tool:

1. Go to the directory that contains the Oracle Process Manager and Notification Server command-line tool, located in `/u02/local/oracle/config/BIInstance1/bin`.
2. Run the `opmnctl` command to start the Oracle Business Intelligence system components:
   - `./opmnctl startall`: Starts Oracle Process Manager and Notification Server and all Oracle Business Intelligence system components
   - `./opmnctl start`: Starts Oracle Process Manager and Notification Server only
   - `./opmnctl startproc ias-component=`*`component_name`*: Starts a particular system component. For example, where `coreapplication_obips1` is the Presentation Services component:

     ```
     ./opmnctl startproc ias-component=coreapplication_obips1
     ```
3. Check the status of the Oracle Business Intelligence system components:

   ```
   ./opmnctl status
   ```

13.5.8.3 Validating Oracle Business Intelligence URLs

Access the following URLs:

- Access `http://BIVH2:10217/analytics` to verify the status of `bi_server2`.
- Access `http://BIVH2:10217/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://BIVH2:10217/xmlpserver` to verify the status of the Oracle BI Publisher application.
- Access `http://BIVH2:10217/ui` to verify the status of the Oracle Real-Time Decisions application.
- Access `http://BIVH2:10217/mapviewer` to verify the status of the map view functionality in Oracle BI EE.
- Access `http://BIVH2:10217/hr` to verify Financial Reporting.
- Access `http://BIVH2:10217/calcmgr/index.htm` to verify Calculation Manager.
- Access `http://BIVH2:10217/aps/Test` to verify APS.
- Access `http://BIVH2:10217/workspace` to verify workspace.

13.5.9 Validating Access Through Oracle HTTP Server

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

1. While `bi_server2` is running, stop `bi_server1` using the Oracle WebLogic Server Administration Console.
2. Access the following URLs to verify that routing and failover is functioning properly:
   - `http://WEBHOST1:10621/analytics`
   - `http://WEBHOST1:10621/xmlpserver`
   - `http://WEBHOST1:10621/ui`
   - `http://WEBHOST1:10621/hr`
   - `http://WEBHOST1:10621/calcmgr/index.htm`
   - `http://WEBHOST1:10621/aps/Test`
   - `http://WEBHOST1:10621/workspace`
3. Start `bi_server1` from the Oracle WebLogic Server Administration Console.
4. Stop `bi_server2` from the Oracle WebLogic Server Administration Console.
5. Access the following URLs to verify that routing and failover is functioning properly:
   - `http://WEBHOST1:10621/analytics`
   - `http://WEBHOST1:10621/xmlpserver`
   - `http://WEBHOST1:10621/ui`
   - `http://WEBHOST1:10621/hr`
   - `http://WEBHOST1:10621/calcmgr/index.htm`
   - `http://WEBHOST1:10621/aps/Test`
   - `http://WEBHOST1:10621/workspace`

13.5.10 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 requires the use of certificates for

the different addresses communicating with the Administration Server and other servers. See Chapter 6, "Configuring Node Manager" for further details. The procedures in that chapter must be performed twice using the information provided in Table 13–3.

Table 13–3 Details for Host Name Verification for Node Manager and Servers

Run	Host Name (Host)	Server Name (WLS_SERVER)
Run1:	*CRMHOST1*	bi_server1
Run2:	*CRMHOST2*	bi_server2

13.5.11 Configuring Server Migration for the Managed Servers

Server Migration is required for proper failover of the Oracle BI Publisher components in the event of failure in any of the *CRMHOST1* and *CRMHOST2* nodes. For more information, see Chapter 16, "Configuring Server Migration."

13.6 Configuring Oracle Essbase Clustering Using the Essbase Failover Automation Tool

This section describes how to set up Essbase clustering in high-availability environments using the `essfoconfig.sh` failover automation tool.

13.6.1 Prerequisites

Do the following before you run the failover automation tool:

1. Make sure the adminHost property in `/u02/local/oracle/config/BIInstancen/config/OPMN/opmn/instance.properties` file reflects the *BIADMINVH* hostname.
2. Create a new wallet with a self-signed certificate for setting up Oracle Process Manager and Notification Server clustering with SSL enabled:
   a. On *CRMHOST1* and *CRMHOST2*, rename the original wallets. For example:

      For *CRMHOST1*:

      ```
      $ mv /u02/local/oracle/config/BIInstance/config/OPMN/opmn/wallet/ /u02/
      local/oracle/config/BIInstance/config/OPMN/opmn/wallet_orig
      ```

      For *CRMHOST2*:

      ```
      $ mv /u02/local/oracle/config/BIInstance1/config/OPMN/opmn/wallet/ /u02/
      local/oracle/config/BIInstance1/config/OPMN/opmn/wallet_orig
      ```

   b. Ensure that the `/u02/local/oracle/config/BIInstancen/config/OPMN/opmn/opmn.xml` file has `ssl enabled="true"` on each node.
   c. Stop Oracle Process Manager and Notification Server:

      For *CRMHOST1*:

      ```
      $ cd /u02/local/oracle/config/BIInstance/bin

      $ ./opmnctl stopall
      ```

      For *CRMHOST2*:

      ```
      $ cd /u02/local/oracle/config/BIInstance1/bin
      ```

```
$ ./opmnctl stopall
```

d. On *CRMHOST1*, do the following:

– Create an Oracle wallet:

```
$ setenv JAVA_HOME ORACLE_BASE/products/fusionapps/jdk6

$ cd ORACLE_BASE/products/fusionapps/oracle_common/bin

$ ./orapki wallet create -wallet wallet location
```

For example

```
$ ./orapki wallet create -wallet /u02/local/oracle/config/BIInstance/
  config/OPMN/opmn/wallet
```

– Enter the wallet password.

– Add a root certificate to an Oracle wallet:

```
$ ./orapki wallet add -wallet wallet location -dn certificate_dn
-keysize
512|1024|2048 -self_signed -validity number of days
```

For example:

```
$ ./orapki wallet add -wallet
/u02/local/oracle/config/BIInstance/config/OPMN/opmn/wallet
-dn 'CN=\"Self-Signed Certificate for
cluster\",OU=OAS,O=ORACLE,L=REDWOODSHORES,ST=CA,C=US' -keysize
1024 -self_signed -validity 3650
```

– Enter the wallet password.

– Create an Oracle wallet with auto-login enabled:

```
$ ./orapki wallet create -wallet wallet location -auto_login
```

For example:

```
$ ./orapki wallet create -wallet
/u02/local/oracle/config/BIInstance/config/OPMN/opmn/wallet -auto_login
```

– Enter the wallet password.

Performing the tasks listed in Step d creates two files in the wallet directory: `cwallet.sso` and `ewallet.p12`.

e. Copy the new wallet directory to *CRMHOST2*.

f. On each node, push the wallet to the AdminServer domain using the `./opmnctl updateinstanceregistration` command:

For *CRMHOST1*:

Update the `adminUsername` entry in the `/u02/local/oracle/config/BIInstance/config/OPMN/opmn/instance.properties` file to correctly reflect the Weblogic Administrator. For example, update:

```
adminUsername=FUSION_APPS_PROV_PATCH_APPID
```

to

```
adminUsername=FAadmin
```

```
$ cd /u02/local/oracle/config/BIInstance/bin

$ ./opmnctl updateinstanceregistration
```

For *CRMHOST2*:

```
$ cd /u02/local/oracle/config/BIInstance1/bin

$ ./opmnctl updateinstanceregistration
```

This command will ask for an Oracle Weblogic Server Administrator password. The `updateinstanceregistration` command updates information registered on the Administration Server for the Oracle instances. Specifically, the `updateinstanceregistration` command updates the registered Oracle Process Manager and Notification Server remote port, remote host, and wallet from the current Oracle Process Manager and Notification Server settings.

g. Restart the Oracle Process Manager and Notification Server:

For *CRMHOST1*:

```
$ cd /u02/local/oracle/config/BIInstance/bin

$ ./opmnctl startall
```

For *CRMHOST2*:

```
$ cd /u02/local/oracle/config/BIInstance1/bin

$ ./opmnctl startall
```

Note: For information on deploying Essbase in SSL mode, see "SSL for Essbase" in *Oracle Enterprise Performance Management System Security Administration Guide*.

---

3. Create the Essbase System Component on *CRMHOST2*. Refer to the *ORACLE_BASE*/`products/fusionapps/bi/bifoundation/admin/provisioning/readme.txt` file for changes to the `opmnctl createcomponent` parameters for Essbase.

```
$ cd /u02/local/oracle/config/BIInstance1/bin
$ ./opmnctl createcomponent -componentName essbaseserver1
-componentType Essbase -appServerDomainHomeLocation AdminServer Domain Home
-epmClusterName EssbaseCluster Name -jpsConfigSetup
false -epmEssbaseAgentPort 10215 -epmEssbaseServerStartPort 10301
-epmEssbaseServerEndPort 10600
```

Example 13–1 Sample Output

```
$ ./opmnctl createcomponent -componentName essbaseserver1 -componentType Essbase
-appServerDomainHomeLocation ORACLE_BASE/config/domains/CRMHOST1/BIDomain
-epmClusterName Essbase_HA-1 -jpsConfigSetup false -epmEssbaseAgentPort 10215
-epmEssbaseServerStartPort 10301 -epmEssbaseServerEndPort 10600
```

```
Command requires login to weblogic admin server (BIADMINVH.MYCOMPANY.COM):
Creating empty component directories...Done
Provisioning Essbase files for essbaseserver1
Registry essbase seeding started
Cluster name: Essbase_HA-1
Getting registry instance...
Registry essbase seeding complete
Creating new registry file at: /u02/local/oracle/config/BIInstance1/Essbase/
  essbaseserver1/bin/essbase.cfg
Creating Parent directories
Setting ESS_CSS_JVM_OPTION3 value to :ORACLE_BASE/instance/domains/
ORACLE_BASE/config/domains/CRMHOST1/BIDomain/config/fmwconfig/jps-config-jse.xml
Setting DOMAIN_HOME value to :ORACLE_BASE/config/domains/CRMHOST1/BIDomain
Setting agentPort value to :10215
Setting serverPortRange value to :10301-10600
Connecting to JMX URL: service:jmx:t3://BIADMINVH:10201/jndi/
weblogic.management.mbeanservers.domainruntime
Registering essbaseserver1 component
Invoking opmn reload...Done
Command succeeded.
```

4. Make sure to update the `arborPATH` property prior to running the Essbase failover automation tool. Do the following from the `/u02/local/oracle/config/BIInstance/config/foundation/11.1.2.0` directory on *CRMHOST1*:

   a. Run the following to determine the ID for `NAME - Essbase_FA_Cluster`:

      ```
      CRMHOST1> $ ./epmsys_registry.sh view ESSBASE_PRODUCT
      ```

      Example output is:

      ```
      Child 1
      NAME -  Essbase_FA_Cluster
      ID -  4a6fdb8aff5464dbS3469229f131f91e8ef2S7ff5
      TYPE -  CLUSTER
      ```

   b. Using the `Essbase_FA_Cluster` ID from the example in Step a, run the following to determine the ID for `Name - essbaseserver1`:

      ```
      CRMHOST1> $ ./epmsys_registry.sh view \
      #4a6fdb8aff5464dbS3469229f131f91e8ef2S7ff5
      ```

      Example output is:

      ```
      Child 1
      NAME -  essbaseserver1
      ID -  4a6fdb8aff5464dbS3469229f131f91e8ef2S7fef
      TYPE -  ESSBASE_SERVER
      ```

   c. Using the `essbaseserver1` ID from the example in Step b, run the following to view the `arborPATH` value:

      ```
      CRMHOST1> $ ./epmsys_registry.sh view \
      #4a6fdb8aff5464dbS3469229f131f91e8ef2S7fef
      ```

      Example output is:

      ```
      arborPath = ORACLE_BASE/config/BIShared/Essbase/essbaseserver1
      ```

   d. Run the following to update the `arborPath` value to `arborPath = ORACLE_BASE/config/BIShared/Essbase/essbaseserver1.bak`:

```
CRMHOST1> $ ./epmsys_registry.sh updateproperty
\#4a6fdb8aff5464dbS3469229f131f91e8ef2S7fef/@arborPath ORACLE_BASE/
config/BIShared/Essbase/essbaseserver1.bak
```

e. Run the following to view the new `arborPath` property value:

```
$ ./epmsys_registry.sh view \
#4a6fdb8aff5464dbS3469229f131f91e8ef2S7fef
```

13.6.2 Running the Essbase Failover Automation Tool

Do the following to run the Essbase failover automation tool:

1. Grant `execute` on `/u02/local/oracle/config/BIInstance`*n* `/bin/essbase_ha/essfoconfig.sh` script on all nodes.
2. Update the *SHARED_FOLDER* (*ORACLE_BASE*`/config/BIShared/Essbase/`*essbaseserver1*) variable in the `/u02/local/oracle/config/BIInstance`*n*`/bin/essbase_ha/essfoenv.properties` file on each node.
3. From *CRMHOST1*, stop the Essbase System Component:

```
$ cd /u02/local/oracle/config/BIInstance/bin

$ ./opmnctl stopproc ias-component=essbaseserver1
```

4. Run `essfoconfig.sh` with `pushprp 1` on *CRMHOST1* and `pushprp 2` on *CRMHOST2*:

```
CRMHOST1> cd /u02/local/oracle/config/BIInstance/bin/essbase_ha/
CRMHOST1> ./essfoconfig.sh pushprp 1
CRMHOST1> ./essfoconfig.sh pushprp 1
```

> **Note:** You must run the `pushprp` command twice: First to copy the `EssFOConfig.properties` file to the *SHARED_FOLDER*, and second to update the `EssFOConfig.properties` file with *CRMHOST1*-specific values.

```
CRMHOST2> cd /u02/local/oracle/config/BIInstance1/bin/essbase_ha/
CRMHOST2> ./essfoconfig.sh pushprp 2
```

5. Update the following properties in the *ORACLE_BASE*`/config/BIShared/Essbase/essbaseserver1/EssFOConfig.properties` file:

```
SYSTEM_HOST=CRMHOST1
SYSTEM_NODE1=Essbase_FA_Cluster
SYSTEM_HOST1=CRMHOST1
SYSTEM_HOST2=CRMHOST2
SYSTEM_USERNAME=WebLogic_Administrator
```

6. Run `essfoconfig.sh` with the `check` option on *CRMHOST1*:

```
/u02/local/oracle/config/BIInstance/bin/essbase_ha/essfoconfig.sh check
```

Example 13–2 Sample Output

```
./essfoconfig.sh check
Fri Dec 10 11:23:59 PST 2010
Essbase Cluster configuration utility.
```

```
Active user properties file: ORACLE_BASE/config/BIShared/Essbase/essbaseserver1/
  EssFOConfig.properties
Command: check

Environment:
ORACLE_HOME= ORACLE_BASE/products/fusionapps/bi
ORACLE_INSTANCE= /u02/local/oracle/config/BIInstance
JAVA_HOME= ORACLE_BASE/products/fusionapps/jdk6
SHARED_FOLDER= ORACLE_BASE/config/BIShared/Essbase
REGUTIL=/u02/local/oracle/config/BIInstance/config/foundation/11.1.2.0/empsys_
registry.sh

Fetch registry metadata
  Extract the Essbase ID from registry metadata
  The Essbase GUID is: 62d5513d9665ae4353f3a57412cbdf8a90f7fb1
Check if Essbase instance is running.
```

7. Run `essfoconfig.sh` with the `create` option on *CRMHOST1*:

```
/u02/local/oracle/config/BIInstance/bin/essbase_ha/essfoconfig.sh create
```

Example 13–3 Sample Output

```
$ ./essfoconfig.sh create
Fri Dec 10 11:25:50 PST 2010
Essbase Cluster configuration utility.
Active user properties file: ORACLE_BASE/config/BIShared/Essbase/
EssFOConfig.properties
Command: create
Fetch registry metadata
  Extract the Essbase ID from registry metadata
  The Essbase GUID is: 62d5513d9665ae4353f3a57412cbdf8a90f7fb1
Add cluster info using template
  Validate the pre-existence of Cluster from registry metadata
  Fetch registry metadata for a specific ESSBASE_PRODUCT
  Extract the Cluster ID
  The Cluster GUID is:
  Replace properties in template
  Replace Cluster GUID in template

The Component Hierarchy has been created.

Extract the Cluster ID from registry metadata
  Fetch registry metadata for a specific ESSBASE_PRODUCT
  Extract the Cluster ID
  The Cluster GUID is: 62d5513d9665ae43118450de12cd1bf46588000
  Fetch Cluster using GUID from registry metadata
Add cluster node1 info using template
  Replace properties in template
Replace Cluster GUID in template

The Component Hierarchy has been created.

Fetch CLUSTER ID object from registry metadata
  Extract Instance GUID (1) for the cluster
  The Cluster Instance 1 GUID is: 62d5513d9665ae43390672fb12cd1bf7b358000
Add cluster node2 info using template
  Replace properties in template
  Replace Cluster GUID in template

The Component Hierarchy has been created.
```

```
Fetch CLUSTER ID object from registry metadata
  Extract Instance GUID (2) for the cluster
  The Cluster Instance 2 GUID is: 62d5513d9665ae431e64a4c512cd1bf9ecd8000
Essbase Cluster configuration over.
```

8. Run `essfoconfig.sh` with the `dump` option on *CRMHOST1*:

```
/u02/local/oracle/config/BIInstance/bin/essbase_ha/essfoconfig.sh dump
```

Example 13–4 Sample Output

```
$ ./essfoconfig.sh dump
Fri Dec 10 11:28:13 PST 2010
Essbase Cluster configuration utility.
Active user properties file: ORACLE_BASE/config/BIShared/Essbase/essbaseserver1/
  EssFOConfig.properties
Command: dump
Important environment info:
User properties file name: ORACLE_BASE/config/BIShared/Essbase/essbaseserver1/
  EssFOConfig.properties
On successful completion, the following registry metadata files will be created:
    ep.md - Contains all ESSBASE_PRODUCT registry entries.
    ep1.md - Contains registry entries for specific ESSBASE_PRODUCT of interest.
    cl.md - Contains registry entries for CLUSTER of interest.
Found user properties file name: ORACLE_BASE/config/BIShared/Essbase/
  essbaseserver1/EssFOConfig.properties
Fetch registry metadata
  Extract the Essbase ID from registry metadata
  The Essbase GUID is: 62d5513d9665ae4353f3a57412cbdf8a90f7fb1
Extract the Cluster ID from registry metadata
  Fetch registry metadata for a specific ESSBASE_PRODUCT
  Extract the Cluster ID
  The Cluster GUID is: 62d5513d9665ae43118450de12cd1bf46588000
  Fetch Cluster using GUID from registry metadata
  Fetch CLUSTER ID object from registry metadata
  Extract Instance GUID (1) for the cluster
  The Cluster Instance 1 GUID is: 62d5513d9665ae43390672fb12cd1bf7b358000
  Fetch CLUSTER ID object from registry metadata
  Extract Instance GUID (2) for the cluster
  The Cluster Instance 2 GUID is: 62d5513d9665ae431e64a4c512cd1bf9ecd8000
```

9. Run `essfoconfig.sh` with the `esscfg` option on *CRMHOST1*:

```
/u02/local/oracle/config/BIInstance/bin/essbase_ha/essfoconfig.sh esscfg
```

Example 13–5 Sample Output

```
$ ./essfoconfig.sh esscfg
Fri Dec 10 11:29:31 PST 2010
Essbase Cluster configuration utility.
Active user properties file: ORACLE_BASE/config/BIShared/Essbase/essbaseserver1/
  EssFOConfig.properties
Command: esscfg
Updating file: ORACLE_BASE/config/BIShared/Essbase/essbaseserver1/bin/essbase.cfg
  Property FailOverMode not defined in ORACLE_BASE/config/BIShared/Essbase/
  essbaseserver1/bin/essbase.cfg
  Adding it now...
```

10. Run `essfoconfig.sh` with the `opmn 1` option on *CRMHOST1*:

```
/u02/local/oracle/config/BIInstance/bin/essbase_ha/essfoconfig.sh opmn 1
```

This command generates a modified `opmn.xml` file, `opmn_EssHA.xml`, under the `/u02/local/oracle/config/BIInstance/config/OPMN/opmn` directory.

Example 13–6 Sample Output

```
$ ./essfoconfig.sh opmn 1
Fri Dec 10 11:30:31 PST 2010
Essbase Cluster configuration utility.
Active user properties file: ORACLE_BASE/config/BIShared/Essbase/essbaseserver1/
  EssFOConfig.properties
Command: opmn
Buildfile: essopmn.xml

_enable-ess-opmn:
     [copy] Copying 1 file to /u01/app/oracle/instances/BIInstance/bin
     [xlst] Processing /u01/app/oracle/instances/BIInstance/config/OPMN/opmn/opmn_
            EssHA.xml
     [xslt] Loading stylesheet /u01/app/oracle/instances/BIInstance/bin/
            essopmn.xslt
     [xslt] : Warning!
     [xslt]             FOUND: Essbase ias-component with child 'Essbase'
     [xslt]             id:essbaseserver1
     [xslt]             module-id:
     [xslt]             Since entry exists in opmn.xml ...update it
     [xslt]             par-id:instance1
     [xslt]             ancst-id:essbaseserver1
     [xslt]             self-id:essbaseserver1
     [xslt]             child-id:essbaseserver1
     [xslt]             ARG:essclu1
     [xslt] : Warning!
     [xslt]             FOUND: Essbase module with
     [xslt]             id:Essbase
     [xslt]             Since entry exists in opmn.xml ...change the name
     [xslt]             par-id:essbaseserver1
     [xslt]             child-id:Essbase
     [xslt] : Warning!
     [xslt]             FOUND: Essbase ARBORPATH property with
     [xslt]             id:ARBORPATH
     [xslt]             value:/u02/local/oracle/config/BIInstance/Essbase/
                        $COMPONENT_NAME
     [xslt] : Warning!
     [xslt]             FOUND: Essbase process-set AGENT property with
     [xslt]             id:AGENT
     [xslt]             numprocs:1

BUILD SUCCESSFUL
Total time: 1 second
```

11. Add a `<topology>` tag under the `<notification-server>` tag to the `/u02/local/oracle/config/BIInstance/config/OPMN/opmn/opmn_EssHA.xml` file, as shown in Example 13–7. (The lines that you must verify and add are shown in bold.)

Example 13–7 opmn_EssHA.xml File

```
<notification-server interface="any">
      <port local="10203" remote="10204"/>
      <ssl enabled="true" wallet-file="/u02/local/oracle/config/BIInstance/
```

```
       config/OPMN/opmn/wallet"/>
      <topology>
         <nodes list="CRMHOST1:10204,CRMHOST2:10204"/>
      </topology>
</notification-server>
```

In the nodes list, enter the primary and secondary nodes and remote port numbers.

12. Save the original `/u02/local/oracle/config/BIInstance/config/OPMN/opmn/opmn.xml` file and copy the `opmn_EssHA.xml` file to the `/u02/local/oracle/config/BIInstance/config/OPMN/opmn/opmn.xml` file.
13. Repeat Steps 10 through 12 on *CRMHOST2*, with the following command replacing the `essfoconfig.sh` command specified in Step 10:

    ```
    /u02/local/oracle/config/BIInstance1/bin/essbase_ha/essfoconfig.sh opmn 2
    ```

14. Shut down all `opmn` components on *CRMHOST1* and *CRMHOST2*:

    ```
    /u02/local/oracle/config/BIInstancen/bin/opmnctl stopall
    ```

    On *CRMHOST1*, *n* can be `BIInstance`, and on *CRMHOST2* it can be `BIInstance1`.

15. Start all `opmn` components on *CRMHOST1* and *CRMHOST2*:

    ```
    /u02/local/oracle/config/BIInstancen/bin/opmnctl startall
    ```

    On *CRMHOST1*, *n* can be `BIInstance`, and on *CRMHOST2* it can be `BIInstance1`.

To shut down Essbase system components cluster-wide:

```
/u02/local/oracle/config/BIInstance/bin/opmnctl stopproc ias-component=essclu1
process-type=Essbase
```

13.7 Validating the System

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

1. While `bi_server2` is running, stop `bi_server1` using the Oracle WebLogic Server Administration Console.
2. Access the following URLs to verify that routing and failover is functioning properly:
   - `http://WEBHOST1:10621/analytics`
   - `http://WEBHOST1:10621/xmlpserver`
   - `http://WEBHOST1:10621/ui`
   - `http://WEBHOST1:10621/hr`
   - `http://WEBHOST1:10621/workspace`
   - `http://WEBHOST1:10621/calcmgr/index.htm`
   - `http://WEBHOST1:10621/aps/Test`
3. Start `bi_server1` from the Oracle WebLogic Server Administration Console.
4. Stop `bi_server2` from the Oracle WebLogic Server Administration Console.

5. Access the following URLs to verify that routing and failover is functioning properly:
   - `http://WEBHOST1:10621/analytics`
   - `http://WEBHOST1:10621/xmlpserver`
   - `http://WEBHOST1:10621/ui`
   - `http://WEBHOST1:10621/hr`
   - `http://WEBHOST1:10621/workspace`
   - `http://WEBHOST1:10621/calcmgr/index.htm`
   - `http://WEBHOST1:10621/aps/Test`

14

Additional Configuration Procedures for Scaling Out Oracle SOA Suite Server

This chapter describes the additional scaleout steps for `soa_server2` on *`CRMHOST2`*.

> **Note:** The Oracle SOA Suite server uses the Java Message Service (JMS) server. JMS requires a shared file system for its file store and transactional log. Each Oracle SOA Suite managed server in a cluster uses a separate local file system on the shared disk. During a node failure, the Oracle SOA Suite server must be moved to a targeted node in order to run the same server using the exact JMS file store and transaction log. To enable this server migration, each Oracle SOA Suite server must be configured with its own virtual IP, which can be floated on any server where the Oracle SOA Suite server is migrated.

The procedures in this chapter use the Oracle SOA Suite server in the Oracle Fusion Customer Relationship Management domain as an example. You must perform the same procedures for the Oracle SOA Suite servers in all other domains.

> **Note:** For Oracle Fusion Customer Relationship Management, the Oracle SOA Suite virtual IPs for *`CRMHOST1`* and *`CRMHOST2`* are called *`CRMSOAVH1`* and *`CRMSOAVH2`*. For all other domains, replace the first three characters with domain-specific syntax. For *`HCMSOAVH1`*, *`SCMSOAVH1`*, *`ICSOAVH1`*, and so on.

This chapter includes the following topics:

- Section 14.1, "Enabling Virtual IPs on CRMHOST1 and CRMHOST2"
- Section 14.2, "Setting the Listen Address for soa_server1"
- Section 14.3, "Setting the Listen Address for soa_server2"
- Section 14.4, "Updating the FusionVirtualHost_crm.conf Configuration File"
- Section 14.5, "Configuring JMS for the Oracle SOA Suite Server"
- Section 14.6, "Configuring Oracle Coherence for Deploying Composites"
- Section 14.7, "Configuring a Default Persistence Store for Transaction Recovery"
- Section 14.8, "Disabling Host Name Verification for the soa_servern Managed Servers"
- Section 14.9, "Restarting Node Manager on CRMHOST1"

- Section 14.10, "Starting and Validating soa_server1 on CRMHOST1"
- Section 14.11, "Restarting Node Manager on CRMHOST2"
- Section 14.12, "Starting and Validating soa_server2 on CRMHOST2"

Note: Before performing any of the procedures in this chapter, ensure that `soa_server1` is running on *`CRMHOST1`* and `soa_server2` is running on *`CRMHOST2`*.

14.1 Enabling Virtual IPs on CRMHOST1 and CRMHOST2

To enable the virtual IP on Linux:

Note: In this example, `ethX` is the ethernet interface (`eth0` or `eth1`) and `Y` is the index (0, 1, 2, and so on). In addition, the *`CRMSOAVH1`* and *`CRMSOAVH2`* VIPs will be used.

1. On *`CRMHOST1`*:
   a. Run the `ifconfig` command as root:

      ```
      /sbin/ifconfig interface:index IPAddress netmask netmask
      ```

      For example:

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

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

      ```
      /sbin/arping -q -U -c 3 -I interface IPAddress
      ```

      For example:

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

   c. Validate that the address is available by pinging it from another node.

      For example:

      ```
      /bin/ping 100.200.140.206
      ```

2. Repeat Steps a through c on *`CRMHOST2`*.

14.2 Setting the Listen Address for soa_server1

Ensure that you have performed the steps described in Section 14.1, and the scale-out steps described in Section 7.3, Section 7.4, and Section 7.5 before setting the `soa_server1` listen address.

Perform these 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 **soa_server1** in the column of the table. The Setting page for `soa_server1` is displayed.
6. Set the **Listen Address** to *`CRMSOAVH1`*.
7. Click **Save**.
8. Click **Activate Changes**.
9. The changes will not take effect until the `soa_server1` Managed Server is restarted (ensure that Node Manager is up and running):
    a. On the Summary of Servers page, select the **Control** tab.
    b. Select **soa_server1** in the table and then click **Shutdown**.
    c. After the server has shut down, select **soa_server1** in the table and then click **Start**.

14.3 Setting the Listen Address for soa_server2

Ensure that you have performed the steps described in Section 14.1 before setting the `soa_server2` listen address.

Perform these 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 **soa_server2** in the column of the table. The Settings page for `soa_server2` is displayed.
6. Set the **Listen Address** to *`CRMSOAVH2`*.
7. Click **Save**.
8. Click **Activate Changes**.
9. The changes will not take effect until the `soa_server2` Managed Server is restarted (ensure that Node Manager is up and running):
    a. On the Summary of Servers page, select the **Control** tab.
    b. Select **soa_server2** in the table and then click **Shutdown**.
    c. After the server has shut down, select **soa_server2** in the table and then click **Start**.

14.4 Updating the FusionVirtualHost_crm.conf Configuration File

To enable Oracle HTTP Server to route to `soa_cluster`, which contains the `soa_servern` managed servers, you must set the WebLogicCluster parameter to the list of nodes in the cluster.

To set the parameter:

1. On *`WEBHOST1`* and *`WEBHOST2`*, update the WebLogicCluster parameter in the *`ORACLE_BASE`*`/config/CommonDomain_webtier/config/OHS/ohs1/moduleconf/FusionVirtualHost_crm.conf` file to contain a cluster list of virtual *`host`*:*`port`* entries. For example:

```
<LocationMatch /soa-infra>
 SetHandler weblogic-handler
 WebLogicCluster CRMSOAVH1:9024,CRMSOAVH2:9024
</Location>
```

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

```
WEBHOST1> ORACLE_BASE/config/CommonDomain_webtier/bin/opmnctl restartproc
ias-component=ohs1

WEBHOST2> ORACLE_BASE/config/CommonDomain_webtier/bin/opmnctl restartproc
ias-component=ohs1
```

The servers specified in the WebLogicCluster parameters are only important 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 the following:

- **Example 1:** If you have a two-node cluster and then add a third member, you do not need to update the configuration to add the third member. The third member will be discovered 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 would fail to route to the cluster. You must ensure that at least one of the listed nodes is running when you start Oracle HTTP Server.

If you list all the members of the cluster, then you guarantee you can route to the cluster, assuming at least one member is running when Oracle HTTP Server is started. For more information on configuring the Oracle WebLogic Server plug-in, see *Oracle Fusion Middleware Using Web Server 1.1 Plug-Ins with Oracle WebLogic Server*.

14.5 Configuring JMS for the Oracle SOA Suite Server

After *CRMHOST1* has been provisioned, the JMS server and file store are set up and configured for *CRMHOST1*. You now must configure the file store for *CRMHOST2*. Configure the location for all persistence stores to a directory visible from both nodes.

To configure the file store for *CRMHOST2*:

1. Log in to the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Services** node and then click the **Persistence Stores** node.

   The Summary of Persistence Stores page appears.
3. Click **Lock & Edit**.
4. Click **New**, and then **Create File Store**.
5. Enter a name (for example, `SOAJMSFileStore_auto_2`), and a target, `soa_server2`:

   ORACLE_BASE/config/domains/*CRMHOST1*/CRMDomain
6. Click **OK** and **Activate Changes**.

7. In the Domain Structure window, expand the **Services** node and then click the **Messaging > JMS Servers** node.

   The Summary of JMS Servers page appears.

8. Click **Lock & Edit**.

9. Click **New**.

10. Enter a name (for example, `SOAJMSServer_2`), then select **SOAJMSFileStore_auto_2** in the Persistence Store dropdown list.

11. Click **Next**.

12. Select **soa_server2** as the target.

13. Click **Finish** and **Activate Changes**.

14. In the Domain Structure window, expand the **Services** node and then click the **Messaging > JMS Modules** node.

    The JMS Modules page appears.

15. In the Change Center, click **Lock & Edit**.

16. Click **SOAJMSModule** and then click the **Subdeployments** tab.

17. Select **SOAJMSServer** under **Subdeployments**.

18. Add the new `SOAJMSServer_2` as additional targets for the subdeployment.

19. Click **Save** and **Activate Changes**.

14.6 Configuring Oracle Coherence for Deploying Composites

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

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

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

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

Specify the nodes using the `tangosol.coherence.wka n` system property, where *n* is the number for each Oracle HTTP Server. The numbering starts at 1. This numbering must be sequential and must not contain gaps. In addition, specify the host name used by Oracle Coherence to create a cluster through the `tangosol.coherence.localhost` system property. This local host name should be the virtual host name used by the SOA server as the listener addresses (*CRMSOAVH1* and *CRMSOAVH2*). Set this property by adding the `-Dtangosol.coherence.localhost` parameters to the Arguments field of the Oracle WebLogic Server Administration Console's Server Start tab.

Note: *CRMSOAVH1* is the virtual host name that maps to the virtual IP where `soa_server1` is listening (in *CRMHOST1*). *CRMSOAVH2* is the virtual host name that maps to the virtual IP where `soa_server2` is listening (in *CRMHOST2*).

To add the host name used by Oracle Coherence:

1. Log in to the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Environment** node.
3. Click **Servers**.

   The Summary of Servers page appears.
4. Select **soa_server1** (represented as a hyperlink) from the column of the table.

   The Settings page appears.
5. Click **Lock & Edit**.
6. Click the **Server Start** tab.
7. Enter the following for `soa_server1` and `soa_server2` into the **Arguments** field.

   For `soa_server1`, enter the following:

```
-Dtangosol.coherence.wka1=CRMSOAVH1
-Dtangosol.coherence.wka2=CRMSOAVH2
-Dtangosol.coherence.localhost=CRMSOAVH1
-Dtangosol.coherence.localport=8089
-Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
```

   For `soa_server2`, enter the following:

```
-Dtangosol.coherence.wka1=CRMSOAVH2
-Dtangosol.coherence.wka2=CRMSOAVH1
-Dtangosol.coherence.localhost=CRMSOAVH2
-Dtangosol.coherence.localport=8089
-Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
```

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

---

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

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

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

> **Note:** The Oracle Coherence cluster used for deployment uses port 8088 by default. This port can be changed by specifying the `-Dtangosol.coherence.wkaX.port` startup parameter.

14.7 Configuring a Default Persistence Store for Transaction Recovery

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

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

To set the location for the default persistence store:

1. Log in to the Oracle WebLogic Server Administration Console (`http://crminternal.mycompany.com:7777/console`).
2. In the Change Center, click **Lock & Edit**.
3. In the Domain Structure window, expand the **Environment** node and then click the **Servers** node. The Summary of Servers page is displayed.
4. Click the name of the server (represented as a hyperlink) in the 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 will store its data files. The directory structure of the path for *CRMHOST1* is the following:

   CRMHOST1`> `*ORACLE_BASE*`/config/domains/`*CRMHOST1*`/CRMDomain/tlogs`

7. Click **Save** and **Activate Changes**.
8. Restart the Managed Servers to activate the changes (ensure that Node Manager is up and running):
   a. Log in to the Oracle WebLogic Server Administration Console (`http://crminternal.mycompany.com:7777/console`).
   b. In the Summary of Servers screen, select the **Control** tab.
   c. Select **soa_server1** and **soa_server2** in the table and then click **Shutdown**.
   d. Restart the `soa_server1` and `soa_server2` 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 `soa_server1` and `soa_server2` must be able to access this directory.

14.8 Disabling Host Name Verification for the soa_server*n* Managed Servers

This step is required if you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server. By default, Host Name Verification should be set to *None*. If it is not, follow the steps below.

If you have not configured the server certificates, you will receive errors when managing the different WebLogic servers. To avoid these errors, disable host name verification while setting up and validating the topology, and enable it again once the enterprise deployment topology configuration is complete.

To disable Host Name Verification:

1. Log in to Oracle WebLogic Server Administration Console. For example, `http://crminternal.mycompany.com:7777/console`.
2. Click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**.

   The Summary of Servers page appears.
5. Select **soa_server1** (represented as a hyperlink) from the column of the table.

   The Settings page appears.
6. Select the **SSL** tab.
7. Expand the **Advanced** section of the page.
8. Set **Hostname Verification** to **None**.
9. Click **Save**.
10. Repeat Steps 1 through 9 for the `soa_server2` managed server.
11. Save and activate the changes.

14.9 Restarting Node Manager on CRMHOST1

To restart Node Manager on *CRMHOST1*:

1. Stop Node Manager by stopping the process associated with it:
   a. If it is running in the foreground in a shell, simply use CTRL+C.
   b. If it is running in the background in the shell, find the associated process and use the `kill` command to stop it. For example:

```
CRMHOST1> ps -ef | grep NodeManager
orcl 9139 9120 0 Mar03 pts/6 00:00:00/bin/sh ./startNodeManager.sh
```

   c. Run the following command:

```
CRMHOST1>kill -9 9139
```

2. Start Node Manager:

```
CRMHOST1> ORACLE_BASE/products/fusionapps/wlserver_10.3/common/nodemanager/
          CRMHOST1/startNodeManagerWrapper.sh
```

14.10 Starting and Validating soa_server1 on CRMHOST1

To start the `soa_server1` managed server on *CRMHOST1*:

1. Access the Administration Console. For example, `http://crminternal.mycompany.com:7777/console`.
2. Click **Servers**.
3. Open the **Control** tab.
4. Select **soa_server1**.
5. Click **Start**.

To validate the `soa_server1` managed server on *CRMHOST1*:

1. Verify that the server status is reported as "Running" in the Administration Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors.
2. Access `http://CRMSOAVH1:9024/soa-infra` and `http://crminternal.mycompany.com:7777/soa-infra` to verify status of `soa_server1`.

> **Note:** Although the `soa_server1` server may be up, some applications may be in a failed state. Therefore, Oracle recommends verifying the above URLs and watching for errors pertaining each individual application in the server's output file.

14.11 Restarting Node Manager on CRMHOST2

To restart Node Manager on *CRMHOST2*, follow the steps in Section 14.9, "Restarting Node Manager on CRMHOST1."

14.12 Starting and Validating soa_server2 on CRMHOST2

To start the `soa_server2` managed server on *`CRMHOST2`* and ensure that it is configured correctly:

1. From the Administration Console, start the `soa_server2` managed server.
2. Verify that the server status is reported as "Running" in the Administration Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors.
3. Access `http://`*`CRMSOAVH2`*`:9024/soa-infra` and `http://`*`crminternal`*`.mycompany.com:7777/soa-infra`.

Note: Although `soa_server2` server may be up, some applications may be in a failed state. Therefore, Oracle recommends verifying the above URLs and watching for errors pertaining each individual application in the server's output file.

15

Configuring Administration Server High Availability

This chapter describes how to configure and validate the Oracle WebLogic Server Administration Server for high availability.

This chapter includes the following topics:

- Section 15.1, "Enabling Administration Server High Availability"
- Section 15.2, "Oracle HTTP Server Configuration"
- Section 15.3, "Validating the Administration Server"
- Section 15.4, "Manually Failing Over the Administration Server to CRMHOST2"
- Section 15.5, "Failing the Administration Server Back to CRMHOST1"

15.1 Enabling Administration Server High Availability

The Administration Server is a singleton application, so it cannot be deployed in an active-active configuration. By default, the Administration Server is only available on the first installed node. 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 failover. The enterprise deployment architecture in this guide calls for the deploying the Administration Server on a disk shared between the primary node and the secondary node.

The following domains are deployed as part of the Oracle Fusion Customer Relationship Management enterprise deployment implementation:

- Oracle Fusion Applications Domain
- Oracle Fusion Customer Relationship Management Domain
- Oracle Fusion Setup Domain
- Oracle Fusion Financials Domain
- Oracle Fusion Human Capital Management Domain
- Oracle Fusion Supply Chain Management Domain
- Oracle Fusion Incentive Compensation Domain
- Oracle Business Intelligence Domain

The process described in this guide initially deploys the Administration Server in shared storage (`/u01/oracle`) mounted on *CRMHOST1*, and Managed Servers in the local disk (`/u02/local/oracle`).

This section contains the following topics:

- Section 15.1.1, "Enabling Administrative Virtual Host on CRMHOST1"
- Section 15.1.2, "Adding a New Machine in the Oracle WebLogic Server Console"
- Section 15.1.3, "Enabling the Administration Server to Listen on the Virtual IP Address"

15.1.1 Enabling Administrative Virtual Host on CRMHOST1

Note: *CRMADMINVH* is used as a generic name in this chapter. For domain-specific administrative virtual host names, see Table 2–1 in Section 2.6, "IPs and Virtual IPs."

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 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.

To enable a virtual IP Address on *CRMHOST1*:

Note: In a UNIX environment, the command must be run as the root user.

---

1. On *CRMHOST1*, 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@CRMHOST1 ~] # /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 *CRMHOST1*, 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
```

15.1.2 Adding a New Machine in the Oracle WebLogic Server Console

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: `CRMHOST1.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 dropdown list.
11. Click **Finish** to associate the Administration Server with the machine.
12. In the Change Center, click **Activate Changes**.

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

Ensure that you have performed the steps described in Section 15.1.1, "Enabling Administrative Virtual Host on CRMHOST1" before setting the Administration Server listen address.

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 column of the table. The Setting page for AdminServer(admin) is displayed.
6. Set the **Listen Address** to *CRMADMINVH* (domain-specific administrative virtual host).
7. Click **Save**.
8. Click **Activate Changes**.
9. The changes will not take effect until the Administration Server is restarted. Follow these 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**.
10. Set the following environment variable:

```
WLST_PROPERTIES="-Dweblogic.security.SSL.trustedCAKeyStore=ORACLE_
BASE/products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks"
```

11. Start the Administration Server again from the command line. Use the `nmconnect` username and password you specified in the Installation Location Screen in Chapter 4.

```
CRMHOST1> cd ORACLE_BASE/products/fusionapps/wlserver_10.3/common/bin/wslt.sh

CRMHOST1> nmConnect(username='username', password='password',
domainName='domain_name', host='CRMHOST1',port='5556', nmType='ssl',
domainDir='ORACLE_BASE/config/domains/CRMHOST1/domain_name')

CRMHOST1> nmStart('AdminServer')
```

15.2 Oracle HTTP Server Configuration

To configure Oracle HTTP Server:

1. On *WEBHOST1*:
   a. `cd ORACLE_BASE/config/CommonDomain_webtier/config/OHS/ohs1/moduleconf`.
   b. Edit the domain-specific virtual host `config` file. For example:

```
cp FusionVirtualHost_crm.conf FusionVirtualHost_crm.conf.org
```

2. Edit the `FusionVirtualHost_crm.conf` file, adding the Administrative virtual host and port. Example 15–1 shows sample code.

> **Note:** Replace *CRMADMINVH* and port with domain-specific Administrative virtual host and port number.

Example 15–1 Add AdministrativeVirtual Host and Port

```
## Context roots for application em
    <Location /em>
        SetHandler weblogic-handler
        WebLogicCluster CRMADMINVH:port
    </Location>

## Context roots for application console
    <Location /console >
        SetHandler weblogic-handler
```

```
    WebLogicCluster CRMADMINVH:port
</Location>
```

3. Restart Oracle HTTP Server: `cd` to *`ORACLE_BASE`*`/config/CommonDomain_webtier/bin` and enter the following:

```
WEBHOST1> ./opmnctl stopall
WEBHOST1> ./opmnctl startall
```

4. Repeat Steps 1 through 3 on *`WEBHOST2`*.

15.3 Validating the Administration Server

Perform these steps to ensure that the Administration Server and Oracle Enterprise Manager Fusion Middleware Control are properly configured:

1. Ensure that you can access the domain-specific Oracle WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control. For example, for the Oracle Fusion Customer Relationship Management domain:

   `http://crminternal.mycompany.com:7777/console`

   `http://crminternal.mycompany.com:7777/em`

2. Repeat Step 1 for other domain by replacing the domain-specific URL.

15.4 Manually Failing Over the Administration Server to CRMHOST2

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 *`CRMHOST1`* to *`CRMHOST2`*.

15.4.1 Prerequisites

Ensure the following:

- The Administration Server is configured to listen on a domain-specific administrative virtual host, and not on **any** address
- The Administration Server is failed over from *`CRMHOST1`* to *`CRMHOST2`* and the two nodes have the following IPs:
  - *`CRMHOST1`*: 100.200.140.165
  - *`CRMHOST2`*: 100.200.140.205
  - *`CRMADMINVH`*: 100.200.140.206. This is the VIP where the domain-specific Administration Server is running, assigned to `ethX:Y`, available in *`CRMHOST1`* and *`CRMHOST2`*.
  - The domain directory where the Administration Server is running on *`CRMHOST1`* is on shared storage and is mounted from *`CRMHOST2`*

15.4.2 Performing the Failover

The following procedure explains how to fail over the Administration Server to a different node (*`CRMHOST2`*) with the Administration Server still using the same Oracle WebLogic Server machine. (This machine is a logical machine, not a physical one.)

To fail over the Administration Server:

1. Stop the Administration Server.
2. Migrate the IP to the second node:
   a. Run the following command as root on *CRMHOST1* (where `X:Y` is the current interface used by *CRMADMINVH*):

      ```
      CRMHOST1> /sbin/ifconfig ethX:Y down
      ```

   b. Run the following command on *CRMHOST2*:

      ```
      CRMHOST2> /sbin/ifconfig interface:index IP_Address netmask netmask
      ```

      For example:

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

      Note: Ensure that the netmask and interface to be used to match the available network configuration in *CRMHOST2*.

3. Update the routing tables with arping. For example:

   ```
   CRMHOST2> /sbin/arping -b -A -c 3 -I eth0 100.200.140.206
   ```

4. Start the Administration Server on *CRMHOST2* using the procedure in Section 15.1.3.
5. Test access to the Administration Server on *CRMHOST2*:
   a. Ensure that you can access the domain-specific Oracle WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control. For example, for the Oracle Fusion Customer Relationship Management domain, use these URLs:
      - `http://crminternal.mycompany.com:7777/console`
      - `http://crminternal.mycompany.com:7777/em`
   b. Repeat Step a for other domain by replacing the domain-specific URL.

      Note: The Administration Server does not use Node Manager for failing over. After a manual failover, the machine name that appears in the Current Machine field in the Administration Console for the server is *CRMHOST1*, and not the failover machine, *CRMHOST2*. Since Node Manager does not monitor the Administration Server, the machine name that appears in the Current Machine field, is not relevant and you can ignore it.

15.5 Failing the Administration Server Back to CRMHOST1

You also must ensure that you can fail back the Administration Server, that is, stop it on *CRMHOST2* and run it on *CRMHOST1*. To do this, migrate *CRMADMINVH* back to *CRMHOST1* node.

To migrate *CRMADMINVH*:

1. Run the following command on *CRMHOST2*:

   ```
   CRMHOST1> /sbin/ifconfig ethX:Y 100.200.140.206 netmask 255.255.255.0
   ```

Note: Ensure that the netmask and interface to be used match the available network configuration in *CRMHOST1*.

2. Run the following command from *CRMHOST1* to update the routing tables through arping:

```
CRMHOST1> /sbin/arping -b -A -c 3 -I ethZ 100.200.140.206
```

3. Start the Administration Server again on *CRMHOST1* using the procedure in Step 3 in Section 15.1.3.

4. Test access to the Administration Server on *CRMHOST1*:

   a. Ensure that you can access the domain-specific Oracle WebLogic Server Administration Console and Oracle Enterprise Manager Fusion Middleware Control. For example, for the Oracle Fusion Customer Relationship Management domain, use these URLs:

      – `http://crminternal.mycompany.com:7777/console`

      – `http://crminternal.mycompany.com:7777/em`

   b. Repeat Step a for other domain by replacing the domain-specific URL.

16 Configuring Server Migration

This chapter describes how to configure server migration in accordance with enterprise deployment recommendations.

This chapter includes the following topic:

- Section 16.1, "Prerequisite"
- Section 16.2, "Migrating Oracle Fusion Applications Domains"

16.1 Prerequisite

Before migrating Oracle Fusion Applications domains, ensure you have completed the steps in Section 14.1, "Enabling Virtual IPs on CRMHOST1 and CRMHOST2," Section 14.2, "Setting the Listen Address for soa_server1," and Section 14.3, "Setting the Listen Address for soa_server2" for all Managed Servers needing to be migrated.

Note: This prerequisite does not apply to the Oracle Business Intelligence domain.

16.2 Migrating Oracle Fusion Applications Domains

The procedures in this section apply to these domains and applications:

- Oracle SOA Suite in the Oracle Fusion Customer Relationship Management domain
- Oracle Fusion Setup
- Oracle Business Intelligence
- Oracle Fusion Human Capital Management
- Oracle Fusion Supply Chain Management
- Oracle Fusion Financials
- Oracle SOA Suite in the Oracle Incentive Compensation domain

16.2.1 About Configuring Server Migration

The procedures described in this chapter must be performed for various components of the enterprise deployment topology outlined in Section 1.5, "Reference Enterprise Deployment Topology." Variables are used in this chapter to distinguish between component-specific items:

- *<DOMAIN>_MANAGED_SERVER1* and *<DOMAIN>_MANAGED_SERVER2*: these refer to the managed WebLogic servers for the enterprise deployment component
- *CRMHOST1* and *CRMHOST2*: these refer to the host machines for the enterprise deployment component
- *CLUSTER*: this refers to the cluster associated with the enterprise deployment component.

The values to be used to these variables are provided in the component-specific chapters in this guide.

In this enterprise topology, you must configure server migration for the *WLS_SERVER1* and *WLS_SERVER2* managed servers. The *WLS_SERVER1* managed server is configured to restart on *CRMHOST2* should a failure occur. The *WLS_SERVER2* managed server is configured to restart on *CRMHOST1* should a failure occur. For this configuration, the *WLS_SERVER1* and *WLS_SERVER2* servers listen on specific floating IP addresses that are failed over by WebLogic Server migration. Configuring server migration for the WLS managed servers consists of the following steps:

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

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

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

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

1. Create a tablespace called 'leasing'. For example, log on to SQL*Plus as the sysdba user and run the following command:

```
SQL> create tablespace leasing logging datafile
'DB_HOME/oradata/orcl/leasing.dbf'
size 32m autoextend on next 32m maxsize 2048m extent management local;
```

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

```
SQL> create user leasing identified by welcome1;
SQL> grant create table to leasing;
SQL> grant create session to leasing;
SQL> alter user leasing default tablespace leasing;
SQL> alter user leasing quota unlimited on LEASING;
```

3. Create the leasing table using the leasing.ddl script:

a. Copy the leasing.ddl file located in either the *ORACLE_BASE*/products/fusionapps/wlserver_10.3/server/db/oracle/817 or the *ORACLE_BASE*/products/fusionapps/wlserver_10.3/server/db/oracle/920 directory to your database node.

b. Connect to the database as the leasing user.

c. Run the leasing.ddl script in SQL*Plus:

```
SQL> @Copy_Location/leasing.ddl;
```

16.2.3 Creating a Multi-Data Source Using the Oracle WebLogic Server Administration Console

The second step is to create a multi-data source for the leasing table from the Oracle WebLogic Server Administration Console. You create a data source to each of the Oracle RAC database instances during the process of setting up the multi-data source, both for these data sources and the global leasing multi-data source.

Please note the following considerations when creating a data source:

- Make sure that this is a non-XA data source.
- The names of the multi-data sources are in the format of <*MultiDS*>-*rac0*, <*MultiDS*>-*rac1*, and so on.
- Use Oracle's Driver (Thin) Version 9.0.1, 9.2.0, 10, 11.
- Use Supports Global Transactions, One-Phase Commit, and specify a service name for your database.
- Target these data sources to the cluster assigned to the enterprise deployment component (*CLUSTER*; see the component-specific chapters in this guide).
- Make sure the initial connection pool capacity of the data sources is set to 0 (zero). To do this, select **Services**, then **JDBC**, and then **Datasources**. In the Datasources screen, click the **Datasource Name**, then click the **Connection Pool** tab, and enter **0** (zero) in the **Initial Capacity** field.

Creating a Multi-Data Source

Perform these steps to create a multi-data source:

1. In the Domain Structure window in the Oracle WebLogic Server Administration Console, click the **Data Sources** link.
2. Click **Lock & Edit**.
3. Select **Multi Data Source** from the **New** dropdown menu.

   The Create a New JDBC Multi Data Source page is displayed.
4. Enter `leasing` as the name.
5. Enter `jdbc/leasing` as the JNDI name.
6. Select **Failover** as algorithm (default).
7. Click **Next.**
8. Select the cluster assigned to the enterprise deployment component as the target.
9. Click **Next**.
10. Select **non-XA driver** (the default).
11. Click **Next**.

12. Click **Create New Data Source**.
13. Enter `leasing-rac0` as the name. Enter `jdbc/leasing-rac0` as the JNDI name. Enter `oracle` as the database type. For the driver type, select Oracle Driver (Thin) for Oracle RAC Service-Instance connections, Versions 10 and later.

    Note: When creating the multi-data sources for the leasing table, enter names in the format of <*MultiDS*>-rac0, <*MultiDS*>-rac1, and so on.

14. Click **Next**.
15. Deselect **Supports Global Transactions**.
16. Click **Next**.
17. Enter the following for your leasing schema:
    - **Service Name:** The service name of the database.
    - **Database Name:** The Instance Name for the first instance of the Oracle RAC database.
    - **Host Name:** 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:** The port number for the database (1521).
    - **Database User Name:** Enter `leasing`.
    - **Password:** The leasing password.
18. Click **Next**.
19. Click **Test Configuration** and verify that the connection works.
20. Click **Next**.
21. Target the data source to the cluster assigned to the enterprise deployment component (*CLUSTER*).
22. Click **Finish**.
23. Click **Create a New Data Source** for the second instance of your Oracle RAC database, target it to the cluster assigned to the enterprise deployment component (*CLUSTER*), repeating the steps for the second instance of your Oracle RAC database.
24. Add `leasing -rac0` and `leasing -rac1` to your multi-data source.
25. Click **Finish**, then **Activate Changes**.

16.2.4 Editing Node Manager's Properties File

The third step is to edit Node Manager's properties file. This needs to be done for the node managers in both nodes where server migration is being configured:

```
Interface=eth0
NetMask=255.255.255.0
UseMACBroadcast=true
```

- **Interface:** This property specifies the interface name for the floating IP (for example, `eth0`).

Do not specify the sub-interface, such as `eth0:1` or `eth0:2`. This interface is to be used without `:0` or `:1`. Node Manager's scripts traverse the different :*X*-enabled IPs to determine which to add or remove. For example, the valid values in Linux environments are `eth0`, `eth1`, `eth2`, `eth3`, `eth`*n*, depending on the number of interfaces configured.

- **NetMask:** This property specifies the net mask for the interface for the floating IP. The net mask should the same as the net mask on the interface; 255.255.255.0 is used as an example in this document.
- **UseMACBroadcast:** This property specifies whether or not to use a node's MAC address when sending ARP packets, that is, whether or not to use the -`b` flag in the `arping` command.

Verify in Node Manager's output (shell where Node Manager is started) that these properties are being used, or problems may arise during migration. You should see something like this in Node Manager's output:

```
...
StateCheckInterval=500
Interface=eth0
NetMask=255.255.255.0
...
```

> **Note:** The steps below are not required if the server properties (start properties) have been properly set and Node Manager can start the servers remotely.

1. Set the following property in the nodemanager.properties file:
   - **StartScriptEnabled:** Set this property to 'true'. This is required for Node Manager to start the managed servers using start scripts.
2. Start Node Manager on *CRMHOST1* and *CRMHOST2* by running the `startNodeManagerWrapper.sh` script, which is located in the *ORACLE_BASE*`/products/fusionapps/wlserver_10.3/common/nodemanager/CRMHOST1` and `ORACLE_BASE/products/fusionapps/wlserver_10.3/common/nodemanager/CRMHOST2` directories.

16.2.5 Setting Environment and Superuser Privileges for the wlsifconfig.sh Script

The fourth step is to set environment and superuser privileges for the `wlsifconfig.sh` script:

1. Ensure that your PATH is set with the environment variables in the terminal from where Node Manager is started, and that it includes these files:

Table 16–1 Files Required for the PATH Environment Variable

File	Located in this directory
wlsifconfig.sh	`/u02/local/oracle/config/domains/`*CRMHOSTn*`/`*ManagedServer*`_Domain/bin/server_migration`
wlscontrol.sh	*ORACLE_BASE*`/products/fusionapps/wlserver_10.3/common/bin`
nodemanager.domains	*ORACLE_BASE*`/products/fusionapps/wlserver_10.3/common/nodemanager/CRMHOST`*n*

2. Grant sudo configuration for the wlsifconfig.sh script.
   - Configure sudo to work without a password prompt.
   - For security reasons, sudo should be restricted to the subset of commands required to run the wlsifconfig.sh script. For example, perform these steps to set the environment and superuser privileges for the wlsifconfig.sh script:
     a. Grant sudo privilege to the WebLogic user ('oracle') with no password restriction, and grant execute privilege on the /sbin/ifconfig and /sbin/arping binaries.
     b. Make sure the script is executable by the WebLogic user ('oracle'). The following is an example of an entry inside /etc/sudoers granting sudo execution privilege for `oracle` and also over `ifconfig` and `arping`:

```
oracle ALL=NOPASSWD: /sbin/ifconfig,/sbin/arping
```

Note: Ask the system administrator for the sudo and system rights as appropriate to this step.

16.2.6 Configuring Server Migration Targets

The fifth step is to configure server migration targets. You first assign all the available nodes for the cluster's members and then specify candidate machines (in order of preference) for each server that is configured with server migration. Follow these steps to configure cluster migration in a migration in a cluster:

1. Log in to the Oracle WebLogic Server Administration Console. For example, For example, `http://crminternal.mycompany.com:7777/console`.
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 (***CLUSTER***) in the Name column of the table.
4. Click the **Migration** tab.
5. Click **Lock & Edit**.
6. In the **Available** field, select the machine to which to allow migration and click the right arrow. In this case, select ***CRMHOST1*** and ***CRMHOST2***.

Note: When there are three (3) hosts, for example *`CRMHOST3`*, select all three hosts.

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**.

Tip: Click **Customize this table** in the Summary of Servers page and move Current Machine from the Available window to the Chosen window to view the machine on which the server is running. This will be different from the configuration if the server gets migrated automatically.

b. Select the server for which you want to configure migration.

c. Click the **Migration** tab, and then click **Lock & Edit**.

d. In the **Available** field, located in the Migration Configuration section, select the machines to which to allow migration and click the right arrow. For ***WLS_SERVER1***, select ***CRMHOST2***. For ***WLS_SERVER2***, select ***CRMHOST1***.

Note: If there are three (3) hosts (*CRMHOST1*, *CRMHOST2*, and *CRMHOST3*) and three (3) servers (*WLS_SERVER1*, *WLS_SERVER2*, and *WLS_SERVER3*), do the following:

In the **Configuration** section, select the machines to which you want to allow migration and click the right arrow. For *WLS_SERVER1*, select *CRMHOST2*; for *WLS_SERVER2*, select *CRMHOST3*; and for *WLS_SERVER3*, select *CRMHOST1*.

---

e. Select **Automatic Server Migration Enabled**. This enables Node Manager to start a failed server on the target node automatically.

f. Click **Save**.

g. Click **Activate Changes**.

h. Restart the administration server, node managers, and the servers for which server migration has been configured.

16.2.7 Testing the Server Migration

The sixth and final step is to test the server migration. Perform these steps to verify that server migration is working properly:

From CRMHOST1:

1. Stop the *WLS_SERVER1* managed server. To do this, run this command:

   ```
   CRMHOST1> 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:

   ```
   CRMHOST1> ps -ef | grep WLS_SERVER1
   ```

2. Watch the Node Manager console. You should see a message indicating that *WLS_SERVER1*'s floating IP has been disabled.

3. Wait for Node Manager to try a second restart of *WLS_SERVER1*. It waits for a fence period of 30 seconds before trying this restart.

4. Once Node Manager restarts the server, stop it again. Node Manager should now log a message indicating that the server will not be restarted again locally.

From CRMHOST2:

1. Watch the local Node Manager console. After 30 seconds since the last try to restart *WLS_SERVER1* on CRMHOST1, Node Manager on CRMHOST2 should prompt that the floating IP for *WLS_SERVER1* is being brought up and that the server is being restarted in this node.
2. As an example, access the soa-infra console in the same IP.

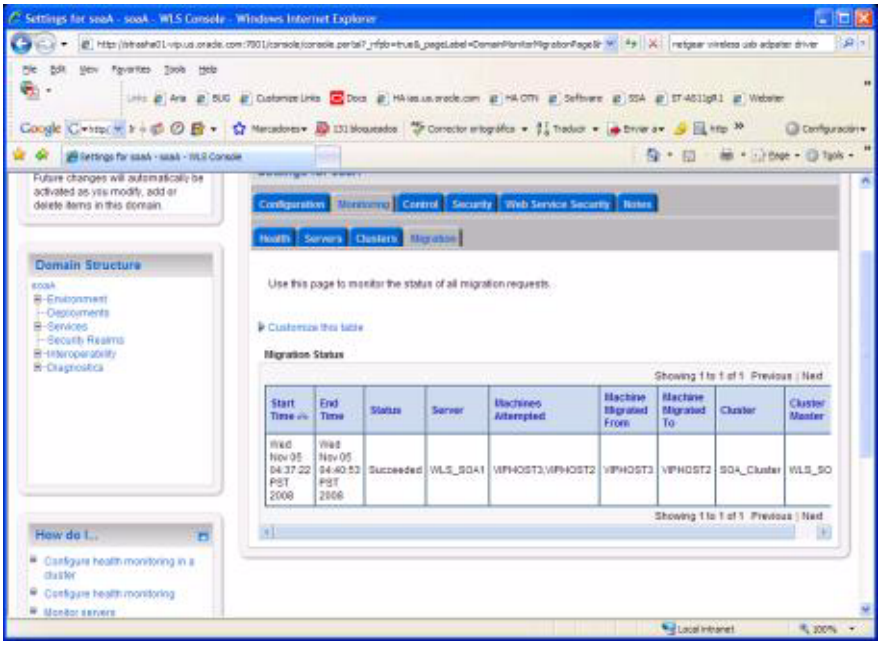
Verification from the Administration Console

Migration can also be verified in the Administration Console:

1. Log in to the Administration Console.
2. Click **Domain** on the left console.
3. Click the **Monitoring** tab and then the **Migration** subtab.

   The Migration Status table provides information on the status of the migration (Figure 16–1).

Figure 16–1 Migration Status Screen in the Administration Console



17

Configuring Oracle Business Intelligence Applications

Configuration of Oracle Business Intelligence Applications is an extension of the existing Oracle Business Intelligence domain. This chapter discusses how the different components of Oracle BI Applications can be installed and configured for high availability.

This chapter includes the following topics:

- Section 17.1, "Introduction to Oracle BI Applications for Oracle Fusion Customer Relationship Management"
- Section 17.2, "Preparing for an Oracle BI Applications Installation"
- Section 17.3, "Performing Additional Configuration Tasks"
- Section 17.4, "Configuring Oracle HTTP Server for the Managed Server"
- Section 17.5, "Performing Additional Data Warehouse Administration Console Tasks"
- Section 17.6, "Validating Oracle BI Applications Components URLs"

17.1 Introduction to Oracle BI Applications for Oracle Fusion Customer Relationship Management

Oracle Fusion Customer Relationship Management is seamlessly integrated with Oracle Business Intelligence Suite to address the full range of analytical requirements. The suite consists of two products, Oracle Transactional Business Intelligence and Oracle Business Intelligence Applications (Oracle BI Applications).

Oracle Transactional Business Intelligence delivers up-to-the minute analysis of a wide range of Oracle Fusion Customer Relationship Management subject areas, whereas Oracle BI Applications provides a more comprehensive historical perspective for Oracle Fusion Customer Relationship Management that is suited to deeper analytical assessments. The product suite is meant to work together to provide customers with the ability to adapt to the rapidly changing and diverse analytical needs required by the business. For example, a customer could use the Transactional Business Intelligence CRM Sales analysis area to view up to the minute pipeline analysis near key forecasting time periods; the projected forecast could then be further validated leveraging the CRM analysis area in Oracle BI Applications to see how this compares vs. previous time periods, sales people, customers, industries, etc. enabling sales management to make adjustments that deliver a more accurate overall forecast.

Oracle Transactional Business Intelligence is an integrated product of Oracle Fusion Applications. Oracle BI Applications is an optional product that you may choose to deploy. The ETL tier of Oracle BI Applications consists of the following components:

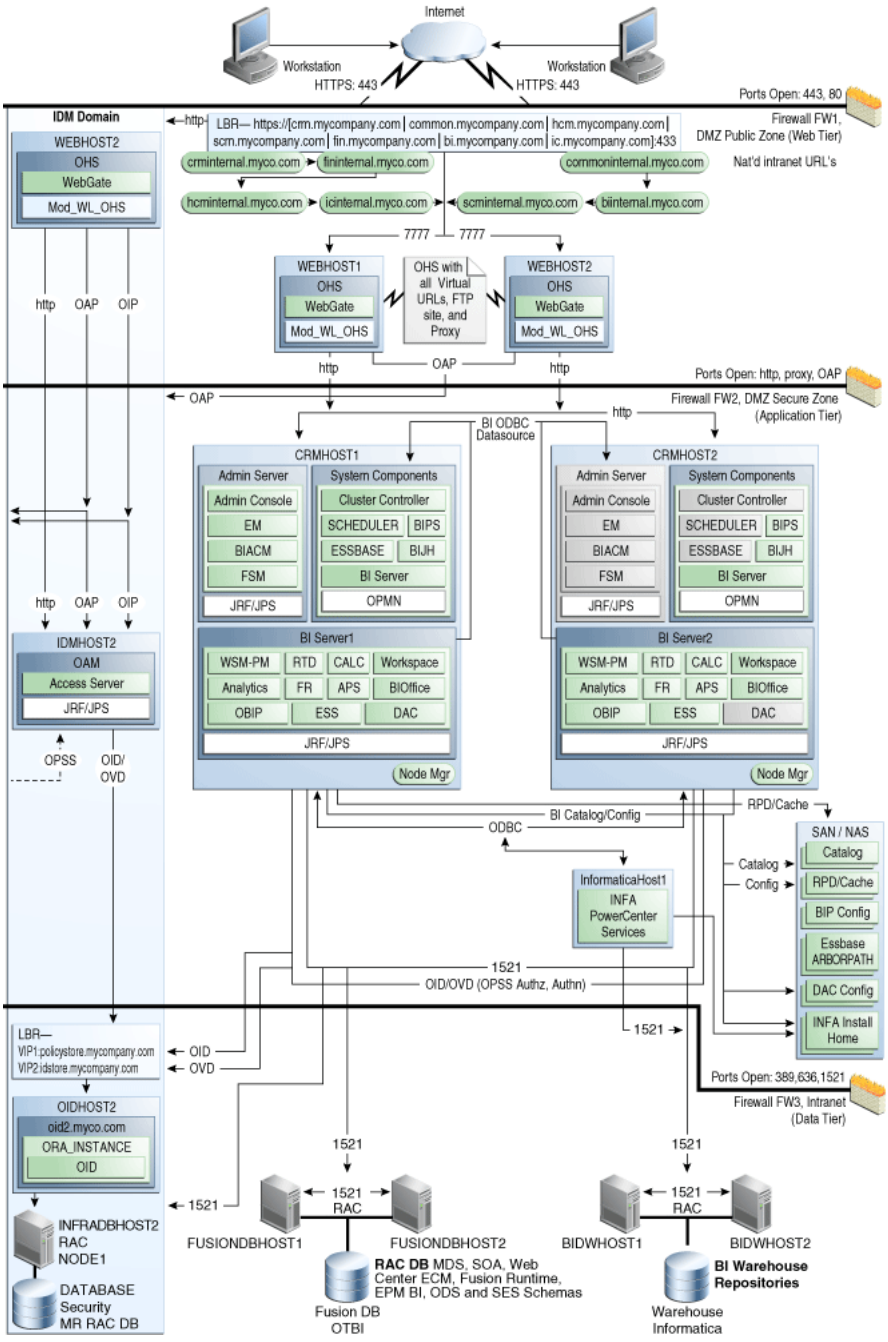
- a Data Warehouse database
- Informatica ETL suite
- Data Warehouse Administration Console (DAC)

The following sections provide more details about implementing these components of Oracle BI Applications for Oracle Fusion Applications.

17.1.1 Topology

Figure 17–1 shows the topology that represents Oracle BI Applications implementation in the Oracle Fusion Applications environment.

Figure 17–1 Data Warehouse for Oracle BI Applications



17.2 Preparing for an Oracle BI Applications Installation

This section describes the following high-level tasks that are required to install Oracle BI Applications:

1. Create the Data Warehouse database.
2. Run the Repository Creation Utility (RCU) to create the Oracle BI Applications schemas for the Data Warehouse.
3. Install and configure Informatica PowerCenter.

For more information, see "Installing and Setting Up Informatica PowerCenter" in *Oracle Fusion Middleware Configuration Guide for Oracle Business Intelligence Applications*.

4. Extend the Oracle Business Intelligence domain by deploying Oracle BI Applications Configuration Manager, Oracle Fusion Functional Setup Manager, and Oracle Business Intelligence Data Warehouse Administration Console (DAC).
5. Perform any necessary post-installation steps to complete the Oracle BI Applications setup.

17.2.1 Creating Databases for Oracle Business Intelligence Applications Components

Before you install Oracle BI Applications, the Data Warehouse Administration Console (DAC), and Informatica PowerCenter, create an Oracle RAC database on *BIDWHOST1* and *BIDWHOST2* to hold the following:

- DAC Repository
- Informatica Domain Configuration Database
- Informatica Repository
- Oracle Business Analytics Warehouse

For information, see the following:

- Section 3.2, "Setting Up the Database."
- "Pre-installation and Pre-deployment Requirements for Oracle BI Applications" in *Oracle Fusion Middleware Configuration Guide for Oracle Business Intelligence Applications*.

17.2.2 Running Oracle BI Applications RCU to Create the Oracle BI Applications Schemas for the Data Warehouse

You must run the Oracle BI Applications Repository Creation Utility (RCU) to create the following Oracle BI Applications schemas:

- Oracle Data Warehouse Administration Console
- Oracle Business Analytics Warehouse

> **Note:** Before running Oracle BI Applications RCU, you must copy the export dump files from the *RCU_HOME*/rcu/integration/biapps/schema directory to the *BIDWHOST1* and *BIDWHOST2* Oracle RAC database nodes. These dump files will be required when entering values in the Custom Variables screen (Figure 17–5). The directory should have read/write access since logs are written to it during the import.

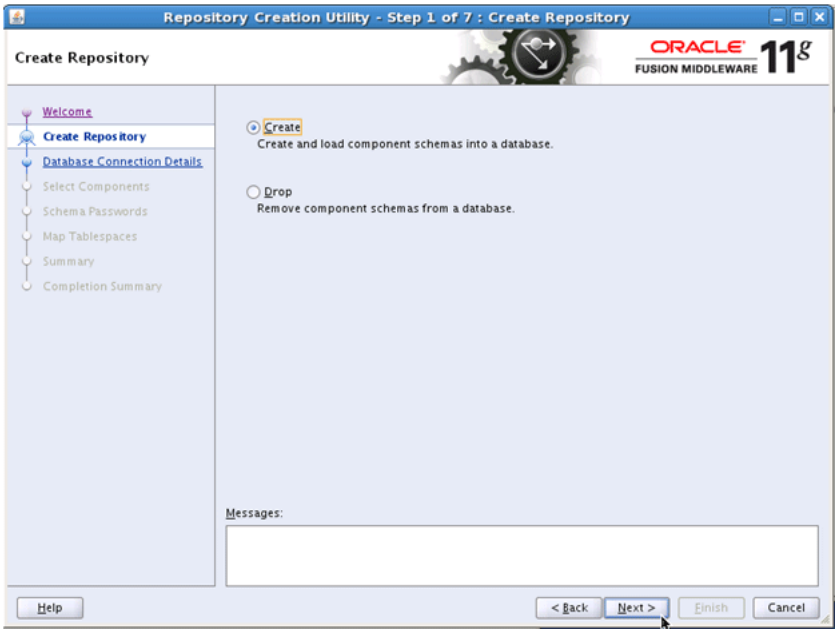
For more information, see "Create the Oracle BI Applications Schemas Using RCU" in *Oracle Fusion Middleware Configuration Guide for Oracle Business Intelligence Applications*.

1. Unzip the *ORACLE_BASE*/repository/installers/biapps_rcu/linux/rcuHomeBIApps.zip file in the RCU home directory, and then start RCU from the bin directory in the RCU home directory:

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

2. In the Welcome screen (if displayed), click **Next**.
3. In the Create Repository screen, shown in Figure 17–2, select **Create** to load component schemas into a database. Click **Next**.

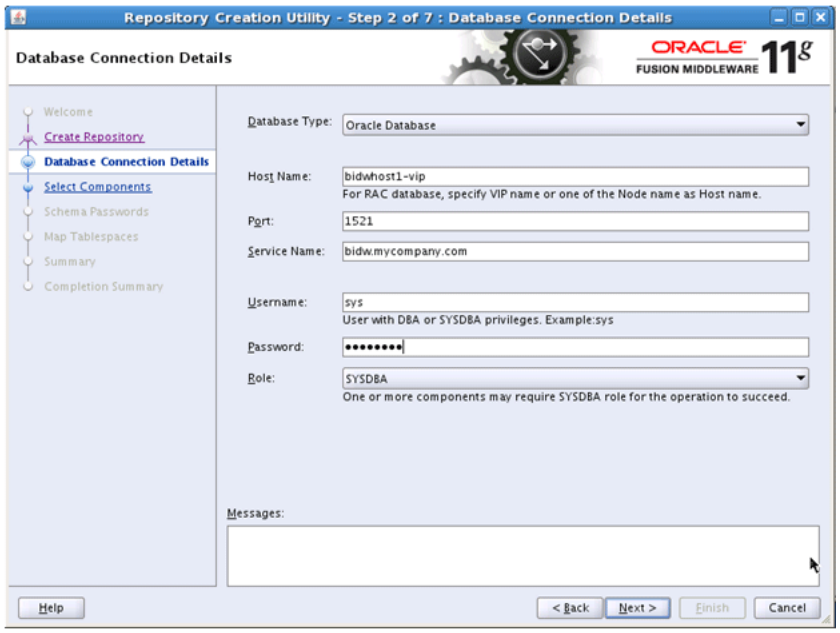
Figure 17–2 Create Repository Screen



4. In the Database Connection Details screen, shown in Figure 17–3, enter connect information for your database:
   - **Database Type**: Select **Oracle Database** from the dropdown list
   - **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: `BIDWHOST1-VIP`
   - **Port**: Specify the listen port number for the database
   - **Service Name**: Specify the service name of the database (`bidw.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 dropdown list: `SYSDBA` (required by the SYS user).

   Click **Next**.

Figure 17–3 Database Connection Details Screen



5. In the Select Components screens, shown in Figure 17–4, do the following:

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

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

   b. First select **Oracle Application Components** and then select the following:

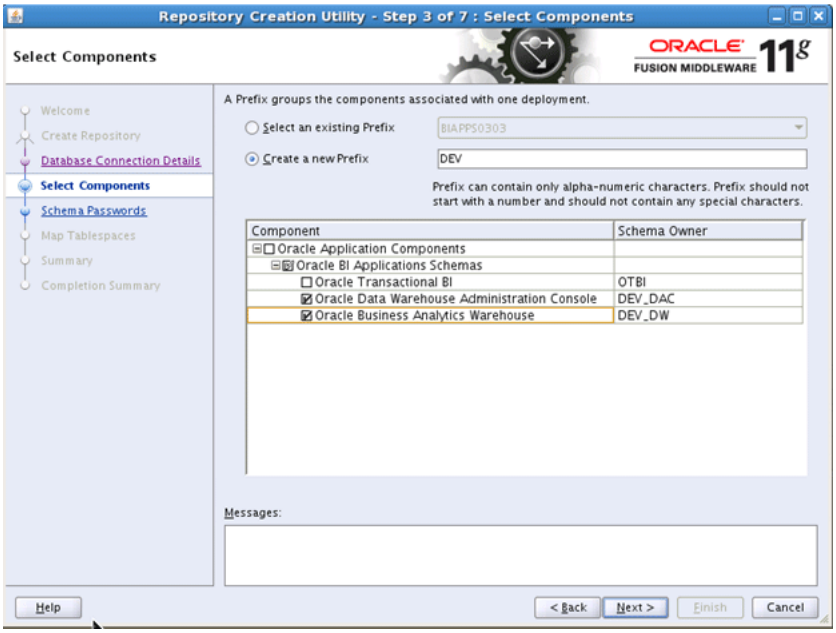
      - **Oracle BI Applications Schemas**

         - **Oracle Data Warehouse Administration Console**

         - **Oracle Business Analytics Warehouse**

   Click **Next**.

Figure 17–4 Select Components Screen

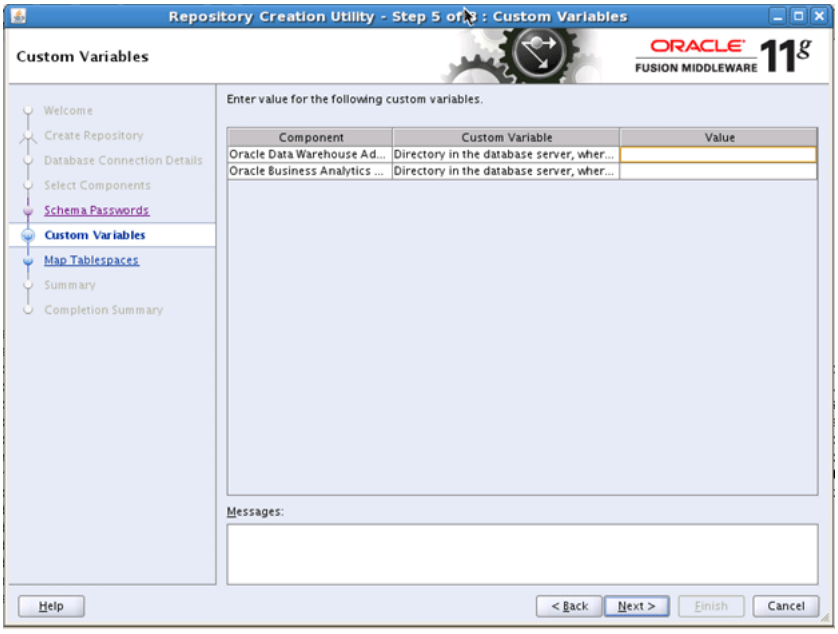


6. In the Schema Passwords screen, enter passwords for the main and additional (auxiliary) schema users, and click **Next**.

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

7. In the Custom Variables screen, shown in Figure 17–5, enter the required values.

Figure 17–5 Custom Variables Screen



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

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

17.2.3 Installing and Configuring Informatica PowerCenter Services

Informatica is a third-party component that is required for Oracle BI Applications implementation. *INFA_HOME*, which refers to the user-specified Informatica installation directory on the machine that hosts Informatica PowerCenter Services, must be on shared storage that is accessible from *CRMHOST1* and *CRMHOST2* where DAC Server will be configured.

Download the Informatica PowerCenter software. For information about how to do so, see "Installing and Setting Up Informatica PowerCenter" in *Oracle Fusion Middleware Configuration Guide for Oracle Business Intelligence Applications*. During installation of Informatica PowerCenter, use the Oracle RAC database installed in Section 17.2.1, "Creating Databases for Oracle Business Intelligence Applications Components" when prompted.

17.2.4 Extending the Oracle Business Intelligence Domain by Deploying Oracle BI Applications Configuration Manager, Functional Setup Manager, and DAC

This section includes the following topics:

- How to Configure DAC, Oracle BI Applications Configuration Manager, and Functional Setup Manager
- Configuring Data Warehouse Administration Console for High Availability

> **Note:** The DAC, Oracle BI Applications Configuration Manager, and Oracle Fusion Functional Setup Manager, configurations are an extension of the existing Oracle Business Intelligence domain. These procedures assume that Oracle Business Intelligence has been installed and configured during the Oracle Fusion Applications Provisioning process.

17.2.4.1 How to Configure DAC, Oracle BI Applications Configuration Manager, and Functional Setup Manager

The DAC, Oracle BI Applications Configuration Manager, and Functional Setup Manager configuration is an extension of the existing Oracle Business Intelligence domain. In this extension, the Oracle BI Applications Configuration Manager, and Functional Setup Manager components are administration components and are targeted to the Administration Server. DAC will be targeted to the Oracle Business Intelligence managed server.

To extend the domain:

1. Run the WebLogic Scripting Tool (WLST) script, *ORACLE_BASE*`/products/fusionapps/bi/dwtools/scripts/install_dwtools.py` from *CRMHOST1*. A sample script is shown in Example 17–1.

Example 17–1 Running the WLST Script

```
ORACLE_BASE/products/fusionapps/bi/common/bin/wlst.sh install_dwtools.py
'DOMAIN_HOME'
'INFORMATICA_SERVER_HOME'
'INFORMATICA_DOMAIN_FILE'
```

```
'DW_DB_URL' 'DW_DB_SCHEMA'
'MDS_DB_URL' 'MDS_DB_SCHEMA'
'DAC_DB_URL' 'DAC_SCHEMA'
'DAC_TARGET'
```

where:

- *DOMAIN_HOME* is the path to the Administration Server Domain Home
- *INFORMATICA_SERVER_HOME* is the path to the Informatica Server Home
- *INFORMATICA DOMAIN FILE* is the path to the Informatica `domains.infa` file location
- *DW_DB_URL* is the string; for example,

```
'jdbc:oracle:thin:@(DESCRIPTION=
(ADDRESS_LIST=(LOAD_BALANCE=on)
(ADDRESS=(PROTOCOL=TCP)
(HOST=BIDWHOST1)(PORT=1521))(ADDRESS=
(PROTOCOL=TCP)(HOST=BIDWHOST2)
(PORT=1521)))(CONNECT_DATA=
(SERVICE_NAME=bidw.mycompany.com)))'
```

- *DW_DB_SCHEMA* is the Data Warehouse schema; for example, *prefix*_DW
- *MDS_DB_URL* is the string; for example,

```
'jdbc:oracle:thin:@(DESCRIPTION=
(ADDRESS_LIST=(LOAD_BALANCE=on)
(ADDRESS=(PROTOCOL=TCP)
(HOST=FUSIONDBHOST1)(PORT=1521))
(ADDRESS=(PROTOCOL=TCP)
(HOST=FUSIONDBHOST2)(PORT=1521)))
(CONNECT_DATA=
(SERVICE_NAME=crm.mycompany.com)))'
```

- *MDS_DB_SCHEMA* is the MDS schema; for example, *prefix*_MDS
- *DAC_DB_URL* is the string; for example,

```
'jdbc:oracle:thin:@(DESCRIPTION=
(ADDRESS_LIST=(LOAD_BALANCE=on)
(ADDRESS=(PROTOCOL=TCP)
(HOST=BIDWHOST1)(PORT=1521))(ADDRESS=
(PROTOCOL=TCP)(HOST=BIDWHOST2)
(PORT=1521)))(CONNECT_DATA=
(SERVICE_NAME=bidw.mycompany.com)))'
```

- *DAC_SCHEMA* is the DAC schema; for example, *prefix*_DAC

Note: When prompted, enter the password for each of the following schemas:

- *prefix*_DW
- *prefix*_MDS
- *prefix*_DAC

- *DAC_TARGET* should be set to the managed server name (`bi_server1`) for an enterprise installation or `AdminServer` for a simple installation.

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

2. Run the following Oracle WebLogic Scripting Tool (WLST) script from *CRMHOST1*:

   ORACLE_BASE/products/fusionapps/bi/dac/scripts/copyDACDomainFiles.py

   For example:

   ORACLE_BASE/products/fusionapps/bi/common/bin/wlst.sh copyDACDomainFiles.py '*ORACLE_HOME*' '*DOMAIN_HOME*'

   where

   - *ORACLE_HOME* is the path to Oracle home. For example, *ORACLE_BASE*/products/fusionapps/bi.
   - *DOMAIN_HOME* is the path to the managed server domain home. For example, /u02/local/oracle/config/domains/*CRMHOST1*/BIDomain.

3. Restart the Administrative Server and all Managed Servers.

4. Run the following WLST script on *CRMHOST1*:

   ORACLE_BASE/products/fusionapps/bi/dwtools/scripts/configure_dwtools.py

   For example:

   ORACLE_BASE/products/fusionapps/bi/common/bin/wlst.sh configure_dwtools.py
   '*WEBLOGIC_ADMINISTRATOR*'
   '*WEBLOGIC_ADMIN_SERVER_HOST*'
   '*WEBLOGIC_ADMIN_SERVER_PORT*'

   You will be prompted for the WebLogic Administrator password.

5. Run the following WLST script on *CRMHOST1*:

   ORACLE_BASE/products/fusionapps/bi/dwtools/scripts/configure_rpd.py

   For example:

   ```
   $ cd ORACLE_BASE/products/fusionapps/bi/dwtools/scripts/
   $ ORACLE_BASE/products/fusionapps/bi/common/bin/wlst.sh configure_rpd.py
   'DOMAIN_HOME'
   'DW_DB_URL'
   'DW_DB_SCHEMA'
   'MASTER_BI_INSTANCE_HOME'
   'WEBLOGIC_ADMIN_SERVER_HOST'
   'WEBLOGIC_ADMIN_SERVER_PORT'
   'WEBLOGIC_ADMINISTRATOR'
   ```

6. Update the /u02/local/oracle/config/domains/*CRMHOST1*/BIDomain/bin/setDomainEnv.sh script to include the following:

   ```
   # Set Informatica Environment for DAC Server
   . ${DOMAIN_HOME}/config/dac/dac_env.sh
   ```

7. For the changes to take effect, restart the Managed Servers and the System Components:

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

b. Select **bi_server1** and **bi_server2** in the table and then click **Shutdown**.

c. After the servers have shut down, select **bi_server1** and **bi_server2** in the table and then click **Start**.

d. Run the following commands to restart the Oracle Business Intelligence system components:

```
$ cd /u02/local/oracle/config/BIInstance/bin
$ ./opmnctl restartproc
```

8. Validate the Oracle BI Applications components:

a. Log in to the Administration Server console (`http://biinternal.mycompany.com:7777/console`) and check the health and status of the Data Warehouse Administration Console Server (DACServer).

b. Validate the following DAC URL: `http://BIVH1:10217/DACServer`.

c. Check to ensure that the following files have been created:

ORACLE_BASE/config/domains/*CRMHOST1*/BIDomain/dac/conf-shared/server.properties

ORACLE_BASE/config/domains/jsc-x4440-1.cz.oracle.com/BIDomain/dac/conf-shared/security/repository/cwallet.sso

d. Log in to the database with the DAC schema user name and password and type the following SQL query:

```
SELECT * FROM W_ETL_REPOS WHERE ROW_WID='DACServerURL';
```

e. Check the `VALUE` column of the result.

The default value before configuration is `http://BIVH1:10217/DACServer`. The hostname and port will be updated in Step 1 in Section 17.2.4.2.

17.2.4.2 Configuring Data Warehouse Administration Console for High Availability

The Data Warehouse Administration Console (DAC) Server is a singleton: only one active Oracle DAC Server is used at any given time. The Oracle WebLogic Server Migration feature is used to protect Oracle DAC server from failures. The Oracle WebLogic Managed Server in which Oracle DAC server runs is listening on a virtual IP that gets migrated to another node when the failure occurs.

For more information on server-migration features, see Chapter 16, "Configuring Server Migration."

1. Run the following WLST script to move the DAC configuration files to a new shared location:

ORACLE_BASE/products/fusionapps/bi/dac/scripts/moveDACConfigLocation.py

For example:

```
ORACLE_BASE/products/fusionapps/bi/common/bin/wlst.sh
moveDACConfigLocation.py 'DOMAIN_HOME' 'DAC_SHARED_LOCATION'
```

where

- *DOMAIN_HOME* is the path to the Administration Server domain home.

- *DAC_SHARED_LOCATION* is the DAC shared location. For example, *ORACLE_BASE*/config/BIShared/dac.

2. Run the following command to restart the Administration Server and the Managed Servers:

```
moveDACConfigLocation.py
```

17.3 Performing Additional Configuration Tasks

For more information, see the following sections in *Oracle Fusion Middleware Configuration Guide for Oracle Business Intelligence Applications*:

- "Overview of Tasks for Setting Up Oracle Business Intelligence Applications"
- "Step 11: Configure SSO for Configuration Manager and Functional Setup Manager"

17.4 Configuring Oracle HTTP Server for the Managed Server

To enable Oracle HTTP Server to route to the Data Warehouse Component managed server, you must set the WebLogicHost parameter.

Perform the following steps:

1. Add the following line to the Oracle HTTP Server's /u01/oracle/config/CommonDomain/webtier*n*/config/CommonDomain_webtier/config/OHS/ohs1/moduleconf/FusionVirtualHost_bi.conf file on *WEBHOST1* and *WEBHOST2*:

```
RedirectMatch 301 ^/DACServer$ /DACServer/

RedirectMatch 301 ^/biacm$ /biacm/

# DAC Server
<LocationMatch ^/DACServer/>
 SetHandler weblogic-handler
 WebLogicHost BIVH1
 WebLogicPort 10217
</LocationMatch>

## Context roots for application biacm
<LocationMatch ^/biacm/>
 SetHandler weblogic-handler
 WebLogicCluster BIADMINVH:10201
</LocationMatch>

## Context roots for application fsm
<LocationMatch /setup >
 SetHandler weblogic-handler
 WebLogicCluster BIADMINVH:10201
</LocationMatch>
```

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

```
WEBHOST1> ORACLE_BASE/config/CommonDomain_webtier/bin/opmnctl restartproc
ias-component=ohs1
WEBHOST2> ORACLE_BASE/config/CommonDomain_webtier1/bin/opmnctl restartproc
ias-component=ohs1
```

17.5 Performing Additional Data Warehouse Administration Console Tasks

Perform the following additional tasks:

1. Set the correct DACServer URL in DAC Repository using the SQL statement shown in Example 17–2.

   The DACServerURL should be set to point to the load balancer virtual server.

Example 17–2 Set the DACServer URL

```
SQL> UPDATE "prefix_DAC"."W_ETL_REPOS" SET VALUE =
'http://biinternal.mycompany.com:7777/DACServer' WHERE ROW_WID = 'DACServerURL'
SQL> commit;
```

2. From the *ORACLE_BASE*/config/domains/*CRMHOST1*/BIDomain/dac directory, start the DAC Client using `startclient.sh` and try to configure a new connection to validate the DAC Server setup. For more information, see "Logging into DAC for the First Time" in *Oracle Fusion Middleware User's Guide for Oracle Business Intelligence Data Warehouse Administration Console*.

17.6 Validating Oracle BI Applications Components URLs

To validate, access the following URLs:

- `http://biinternal.mycompany.com:7777/biacm` to verify the status BI Applications Configuration Manager
- `http://biinternal.mycompany.com:7777/setup/faces/TaskListManagerTop` to verify the status of Oracle Fusion Functional Setup Manager

  Also, ensure that clicking on the "Perform Functional Configurations" link from Oracle BI Applications Configuration Manager launches Functional Setup Manager.
- `http://BIVH1:10217/DACServer` to verify the status of the DAC Server

Verify URLs to ensure that appropriate routing is working from the HTTP Server to the DAC Server.

To verify, access `http://WEBHOST1:10621/DACServer` and verify the appropriate functionality.

18 Managing the Topology

This chapter describes some operations that you can perform after you have set up the topology, including monitoring, scaling, and backing up your topology.

This chapter includes the following topics:

- Section 18.1, "Scaling the Topology for Additional Nodes"
- Section 18.2, "Performing Backups and Recoveries"
- Section 18.3, "Monitoring the Topology"
- Section 18.4, "Migrating from a Test Environment to a Production Environment"
- Section 18.5, "Configuring Log File Rotation"
- Section 18.6, "Patching the Topology"
- Section 18.7, "Auditing"
- Section 18.8, "Troubleshooting"

> **Note:** For Oracle Universal Content Management scale out only, use the procedure described in Section 8.4, "Scaling Out Oracle Universal Content Management."

18.1 Scaling the Topology for Additional Nodes

You can scale out and or scale up the enterprise topology. When you scale up the topology, you add new managed servers to nodes that are already running on one or more managed servers. When you scale out the topology, you add new managed servers to new nodes.

This section includes the topics:

- Section 18.1.1, "Scaling Out the Topology (Adding Managed Servers to a New Node) for Oracle ADF Server"
- Section 18.1.2, "Scaling Up the Topology (Adding Managed Servers to an Existing Node) for Oracle ADF Server"
- Section 18.1.3, "Scaling Out the Topology (Adding Managed Servers to a New Node) for Oracle SOA Suite Server"
- Section 18.1.4, "Scaling Up the Topology (Adding Managed Servers to an Existing Node) for Oracle SOA Suite Server"

- Section 18.1.5, "Scaling Out the Topology (Adding Managed Servers to a New Node) for Oracle Business Intelligence"
- Section 18.1.6, "Scaling Up the Topology for Oracle Business Intelligence"

18.1.1 Scaling Out the Topology (Adding Managed Servers to a New Node) for Oracle ADF Server

When scaling out the topology, you add new managed servers configured to new nodes.

> **Note:** The steps provided in this section also can be used to scale out additional hosts, such as *CRMHOST4*, *CRMHOST5*, and so on.

18.1.1.1 Prerequisites for Scaling Out the Topology for Oracle ADF Server

Before you begin, ensure the following:

- Node Manager has been started in the Secure Sockets Layer (SSL) mode by following the instructions in Chapter 6, "Configuring Node Manager"
- You are starting with a clean machine if it is the first time it is being used for a scale out
- The `/etc/hosts` file has proper entries. To verify, ping this machine with the fully qualified name of the machine
- The user created on *CRMHOST3* should the same as the user on *CRMHOST1*
- The directory structure `/u01/oracle` is mounted to same shared file system as *CRMHOST1*
- The directory structure `/u02/local/oracle/config` on *CRMHOST3* has been created
- The initial Oracle Fusion Customer Relationship Management deployment on *CRMHOST1* has already been done and verified by provisioning

18.1.1.2 Adding a New Machine in the Oracle WebLogic Server Console

To add a new machine:

1. Log in to the Administration Server: `http://`*commoninternal*`.mycompany.com:7777/console`.
2. Navigate to **CommonDomain > Environment > Machines**.

   LocalMachine is located in the right-hand pane.
3. In the left-hand pane, click **Lock & Edit**.
4. In the right-hand pane, first click **New** to add the remote machine, and then specify the following:
   - Name - enter *CRMHOST3*
   - Machine operating system - Unix
5. Click **Next**.
6. In the window that opens, set the following attributes:
   - Type - SSL

- Listen Address - <*CRMHOST3*>

Note: The "localhost" default value here is wrong.

- Listen port - 5556

7. Click **Finish** and activate the changes.

Note: If you get an error when activating the changes, see Section 18.8.18, "Administration Console Redirects from Internal URL to Container URL after Activation" for the temporary solution.

18.1.1.3 Packing and Unpacking the Managed Server Domain Home

Since the *CRMHOST1* domain directory file system is also available from *CRMHOST3*, both the `pack` and `unpack` commands can be executed from the *CRMHOST3*.

1. Do the following:
   a. Change directory to *ORACLE_BASE*`/products/fusionapps/oracle_common/common/bin`.
   b. Run the `pack` command. For `CommonDomain`, for example:

```
CRMHOST3> ./pack.sh -managed=true -domain=ORACLE_BASE/config/domains/
CRMHOST1/CommonDomain -template=ORACLE_BASE/user_templates/
CommonDomain_managed.jar -template_name="Common_Managed_Server_Domain"
```

2. Ensure that `/u02/local/oracle/config/domains/`*CRMHOST3*`/`*CommonDomain* is empty, and then run the `unpack` command:

```
CRMHOST3> ./unpack.sh -domain=/u02/local/oracle/config/domains/
CRMHOST3/CommonDomain -template=ORACLE_BASE/user_templates/
CommonDomain_managed.jar
```

   Here, *ORACLE_BASE* is shared, and `/u02/local` is local to *CRMHOST3*.

18.1.1.4 Cloning Managed Servers and Assigning Them to CRMHOST3

To add a managed server and assign it to *CRMHOST3*:

1. Log in to the Administration Server: `http://`*commoninternal*`.mycompany.com:7777/console`.
2. Navigate to **CommonDomain > Environment > Servers**.
3. Switch to **Lock & Edit** mode.
4. Select the ***Managed_Servers*** checkbox (for example, **HomePageServer_1**) and then click **Clone**.
5. Specify the following Server Identity attributes:
   - Server Name - `HomePageServer_3`

Note: To ensure consistency in naming, copy the name of the server shown in **Server Identity** and paste it into the **Server Name** field. Then change the number to "_3"

- Server Listen Address - <*CRMHOST3*>
- Server Listen Port - leave "as is"

6. Click **OK**.

   You now should see the newly cloned server, `HomePageServer_3`.

7. Click **HomePageServer_3** and change the following attributes:
   - Machine - <*CRMHOST3*>
   - Cluster Name - Default, HomePageCluster
8. Click **Save** and then **Activate Changes**.
9. From the **Name** column, click the **HomePageServer_3** scaled-out server link.
10. Click **Lock & Edit**, and then choose the **Keystores** tab.
11. Ensure that the keystores value is **Custom Identity and Custom Trust**.
12. Do the following:
    a. Change the Custom Identity Keystore path to point to the *ORACLE_BASE*/products/fusionapps/wlserver_10.3/server/lib/*CRMHOST3*_fusion_identity.jks file.
    b. Leave the Custom Identity Keystore type blank.
    c. Change the Custom Identity Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."
    d. Re-enter the Confirm Custom Identity Keystore Passphrase.
    e. Ensure that the Confirm Custom Trust Keystore path is pointing to the *ORACLE_BASE*/products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks file.
    f. Leave the Custom Trust Keystore type blank.
    g. Change the Custom Trust Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."
    h. Re-enter the Custom Trust Keystore Passphrase.
    i. Click **Save**.
13. Choose the **SSL** tab.
    a. Make sure that Identity and Trust Locations is set to **Keystores**.
    b. Change the Private Key Alias to *CRMHOST3*_fusion.
    c. Change the Private Key Passphrase to the *keypassword*, as described in the second bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."
    d. Re-enter the *keypassword* from Step c for the Confirm Private Key Passphrase.
    e. Click **Save**.

14. Click **Activate Changes**.
15. Repeat Steps 2 to 14 for all the newly cloned managed servers on this domain.
16. Set the following environment variable on *CRMHOST3*:

```
WLST_PROPERTIES="-Dweblogic.security.SSL.trustedCAKeyStore=ORACLE_BASE/
products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks"
```

17. Restart the domain's Administration Server:

```
CRMHOST3> ORACLE_BASE/products/fusionapps/wlserver_10.3/common/bin/wlst.sh

CRMHOST3> nmConnect(username='<username>', password='<password>',
domainName='CommonDomain', host='CRMHOST1',port='5556',
nmType='ssl', domainDir='ORACLE_BASE/config/domains/CRMHOST1/CommonDomain')

CRMHOST3> nmStart('AdminServer')
```

> **Note:** The *username* and *password* used in the `nmConnect` are the Node Manager credentials (username and password) specified when creating the provisioning plan. This is shown in Figure 4–3 in "Using the Provisioning Process to Install Components for an Enterprise Deployment".

18. Run the newly created managed server:
    a. Navigate to **CommonDomain > Environment > Servers > Control**.
    b. Check the newly created managed server and click **Start**.
    c. Navigate to **CommonDomain > Environment > Servers** and check the **State** to verify that the newly created managed servers are running.
19. Log in to the Administration Server once again (`http://commoninternal.mycompany.com:7777/console`) and verify that all the managed servers, including scaled-out servers, are running.

> **Note:** For all the scaled-up and scaled-out servers, change the Arguments in the `/u02/local/oracle/config/domains/HOSTNAME/DomainName/servers/ManagedServer/data/nodemanager/startup.properties` file to the following:
>
> ```
> Arguments=-DJDBCProgramName\=DS/CommonDomain/HomePageServer_3
> -Dserver.group\=HomePageCluster
> ```

Note: For all the scaled-up and scaled-out managed servers, do the following:

1. Access the Oracle WebLogic Server Administration Console for the CommonlDomain:

   `http://commoninternal.mycompany.com:7777/console`

2. Navigate to **Environment > Servers** and click the "Managed Server" link.
3. First select the **Logging** tab and then the **HTTP** tab.
4. Update the following parameters:

   – Log file name: logs/access.log.%yyyyMMdd%

   – Rotation Type: By Time

   – Limit number of retained files: leave this option unchecked

   – Rotate log file on startup: leave this option unchecked

5. Click **Save**.
6. Expand **Advanced Node** and set the following:

   – Format: Extended

   – Extended Logging Format Fields:

   ```
   date time time-taken cs-method cs-uri sc-status
   sc(X-ORACLE-DMS-ECID) cs(ECID-Context) cs(Proxy-Remote-User)
   cs(Proxy-Client-IP)
   ```

7. Click **Save** and **Activate Changes**.
8. Restart the Managed Server for the changes to take affect.

18.1.1.5 Validating the System

You should verify URLs to ensure that the appropriate routing and failover are working.

To verify the URLs:

1. Log in to the `CommonDomain` Oracle WebLogic Server Administration Console and stop all the managed servers on the *`CRMHOST1`* while the managed servers on *`CRMHOST3`* are running.
2. Access the following URL to verify that routing and failover are functioning properly. (Ensure the log in prompt is visible.)

   `https://commonexternal.mycompany.com/homePage/faces/AtkHomePageWelcome`

3. Log in to the `CommonDomain` Oracle WebLogic Server Administration Console and stop all the managed servers on *`CRMHOST3`*.
4. Start the managed servers on *`CRMHOST1`*.
5. Repeat Step 2. (Ensure the log in prompt is visible.)
6. Start all the managed servers on *`CRMHOST3`* and verify that they are running on *`CRMHOST1`*.

18.1.2 Scaling Up the Topology (Adding Managed Servers to an Existing Node) for Oracle ADF Server

Before performing the procedures in this section, ensure that `CommonDomain` and its managed servers are running.

18.1.2.1 Cloning Managed Servers and Assigning Them to CRMHOST3

To add a managed server and assign it to *`CRMHOST3`*:

1. Log in to the Administration Server: `http://`*`commoninternal`*`.mycompany.com:7777/console`.
2. Navigate to **CommonDomain > Environment > Servers**.
3. Switch to **Lock & Edit** mode.
4. Select the ***Managed_Servers*** checkbox (for example, **HomePageServer_1**) and then click **Clone**.
5. Specify the following Server Identity attributes:
   - Server Name - `HomePageServer_4`

     Note: To ensure consistency in naming, copy the name of the server shown in **Server Identity** and paste it into the **Server Name** field. Then change the number to "_4".

   - Server Listen Address - <*`CRMHOST3`*>
   - Server Listen Port - leave "Give an unused port on the machine *`CRMHOST3`*"
6. Click **OK**.
7. Navigate back to **CommonDomain > Environment > Servers**. You now should see the newly cloned server, `HomePageServer_4`.
8. Click **HomePageServer_4** and change the following attributes:
   - Machine - <*`CRMHOST3`*>
   - Cluster Name - Default, HomePageCluster
9. From **HomePageServer_4**, click **Advanced** and then select the **WebLogic Plug-In Enabled** checkbox.
10. Run the newly created managed server:
    a. Navigate to **CommonDomain > Environment**.
    b. From the **Navigation** pane on the Oracle WebLogic Server console, select **Activate Changes**.
    c. Navigate to **CommonDomain > Environment > Servers > Control**.
    d. Check the newly created managed server and click **Start**.
    e. Navigate to **CommonDomain > Environment > Servers** and check the **State** to verify that the newly created managed servers are running.
11. Log in to the Administration Server once again (`http://`*`commoninternal`*`.mycompany.com:7777/console`) and verify that all the managed servers, including scaled-up servers, are running.

Note: For all the scaled-up and scaled-out servers, change the Arguments in the `/u02/local/oracle/config/domains/HOSTNAME/DomainName/servers/ManagedServer/data/nodemanager/startup.properties` file to the following:

```
Arguments=-DJDBCProgramName\=DS/CommonDomain/HomePageServer_4
-Dserver.group\=HomePageCluster
```

Note: For all the scaled-up and scaled-out managed servers, do the following:

1. Access the Oracle WebLogic Server Administration Console for the CommonDomain:

   `http://commoninternal.mycompany.com:7777/console`

2. Navigate to **Environment > Servers** and click the "Managed Server" link.
3. First select the **Logging** tab and then the **HTTP** tab.
4. Update the following parameters:

   – Log file name: logs/access.log.%yyyyMMdd%

   – Rotation Type: By Time

   – Limit number of retained files: leave this option unchecked

   – Rotate log file on startup: leave this option unchecked

5. Click **Save**.
6. Expand **Advanced Node** and set the following:

   – Format: Extended

   – Extended Logging Format Fields:

   ```
   date time time-taken cs-method cs-uri sc-status
   sc(X-ORACLE-DMS-ECID) cs(ECID-Context) cs(Proxy-Remote-User)
   cs(Proxy-Client-IP)
   ```

7. Click **Save** and **Activate Changes**.
8. Restart the Managed Server for the changes to take affect.

18.1.2.2 Validating the System

You should verify URLs to ensure that the appropriate routing and failover are working.

To verify the URLs:

1. Log in to the `CommonDomain` Oracle WebLogic Server Administration Console and stop the `HomePageServer_1`, `HomePageServer_2`, and `HomePageServer_3` scaled-up managed servers on *`CRMHOST1`*, *`CRMHOST2`*, and *`CRMHOST3`*.
2. Access the following URL to verify that routing and failover are functioning properly. (Ensure the log in prompt is visible.)

   `https://commonexternal.mycompany.com/homePage/faces/AtkHomePageWelcome`

3. Log in to the `CommonDomain` Oracle WebLogic Server Administration Console and stop the `HomePageServer_4` managed server on *`CRMHOST3`*.
4. Start the `HomePageServer_1` managed server on *`CRMHOST1`*.
5. Repeat Step 2. (Ensure the log in prompt is visible.)
6. Start all the managed servers on *`CRMHOST3`* and verify that they are running on *`CRMHOST1`*.

18.1.3 Scaling Out the Topology (Adding Managed Servers to a New Node) for Oracle SOA Suite Server

When scaling out the topology, you add new managed servers configured to new nodes.

18.1.3.1 Prerequisites for Scaling Out the Topology for Oracle SOA Suite Server

Before you begin, ensure the following:

- Node Manager has been started in the Secure Sockets Layer (SSL) mode by following the instructions in Chapter 6, "Configuring Node Manager"
- You are starting with a clean machine if it is the first time it is being used for a scale out
- The `/etc/hosts` file has proper entries. To verify, ping this machine with the fully qualified name of the machine
- The user created on *`CRMHOST3`* should the same as the user on *`CRMHOST1`*
- The directory structure `/u01/oracle` is mounted to same shared file system as *`CRMHOST1`*
- The directory structure `/u02/local/oracle/config` on *`CRMHOST3`* has been created
- The initial Oracle Fusion Customer Relationship Management deployment on *`CRMHOST1`* has already been done and verified by provisioning

18.1.3.2 Adding a New Machine in the Oracle WebLogic Server Console

If you have not already added *`CRMHOST3`*, follow these steps:

1. Log in to the Administration Server: `http://`*`crminternal`*`.mycompany.com:7777/console`.
2. Navigate to **CRMDomain > Environment > Machines**.

   LocalMachine is located in the right-hand pane.
3. In the left-hand pane, click **Lock & Edit**.
4. In the right-hand pane, first click **New** to add the remote machine, and then specify the following:
   - Name - enter *`CRMHOST3`*
   - Machine operating system - Unix
5. Click **Next**.
6. In the window that opens, set the following attributes:
   - Type - SSL

- Listen Address - <*CRMHOST3*>

> **Note:** The "localhost" default value here is wrong.

- Listen port - 5556

7. Click **Finish** and activate the changes.

> **Note:** If you get an error when activating the changes, see Section 18.8.18, "Administration Console Redirects from Internal URL to Container URL after Activation" for the temporary solution.

18.1.3.3 Packing and Unpacking the Managed Server Domain Home

Since the *CRMHOST1* domain directory file system is also available from *CRMHOST3*, both the `pack` and `unpack` commands can be executed from the *CRMHOST3*.

1. Do the following:
   a. Change directory to *ORACLE_BASE*`/products/fusionapps/oracle_common/common/bin`.
   b. Run the `pack` command. For example, for `CRMDomain`:

```
CRMHOST3> ./pack.sh -managed=true -domain=ORACLE_BASE/config/domains/
CRMHOST1/CRMDomain -template=ORACLE_BASE/user_templates/
CRMDomain_managed.jar -template_name="CRM_Managed_Server_Domain"
```

2. Run the `unpack` command:

```
CRMHOST3> ./unpack.sh -domain=/u02/local/oracle/config/domains/
CRMHOST3/CRMDomain -template=ORACLE_BASE/user_templates/CRMDomain_managed.jar
```

   Here, *ORACLE_BASE* is shared, and `/u02/local` is local to *CRMHOST3*.

18.1.3.4 Cloning Managed Servers and Assigning Them to CRMHOST3

To add a managed server and assign it to *CRMHOST3*:

1. Log in to the Administration Server: `http://`*crminternal*`.mycompany.com:7777/console`.
2. Navigate to **CRMDomain > Environment > Servers**.
3. Switch to **Lock & Edit** mode.
4. Select the ***Managed_Servers*** checkbox (for example, **soa_server1**) and then click **Clone**.
5. Specify the following Server Identity attributes:
   - Server Name - `soa_server3`

> **Note:** To ensure consistency in naming, copy the name of the server shown in **Server Identity** and paste it into the **Server Name** field. Then change the number to "_3".

- Server Listen Address - <*CRMHOST3*>
- Server Listen Port - leave "as is"

6. Click **OK**.
7. Navigate back to **CRMDomain > Environment > Servers**. You now should see the newly cloned sales server, `soa_server3`.
8. Click **soa_server3** and change the following attributes:
   - Machine - <*CRMHOST3*>
   - Cluster Name - Default, SOACluster
9. From **soa_server3**, click **Advanced** and then select the **WebLogic Plug-In Enabled** checkbox.
10. Run the newly created managed server:
    a. Navigate to **CRMDomain > Environment**.
    b. From the **Navigation** pane on the Oracle WebLogic Server console, select **Activate Changes**.
    c. Navigate to **CRMDomain > Environment > Servers > Control**.
    d. Check the newly created managed server and click **Start**.
    e. Navigate to **CRMDomain > Environment > Servers** and check the **State** to verify that the newly created managed servers are running.
11. From the **Name** column, select the scaled-out server, `soa_server3`.
12. Click **Lock & Edit**, and then choose the **Keystores** tab.
13. Change the Keystores dropdown list value to the **Custom Identity and Custom Trust** setting.
14. Do the following:
    a. Change the Custom Identity Keystore path to point to the *ORACLE_BASE*/products/fusionapps/wlserver_10.3/server/lib/*CRMHOST3*_fusion_identity.jks file.
    b. Leave the Custom Identity Keystore type blank.
    c. Change the Custom Identity Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."
    d. Re-enter the Confirm Custom Identity Keystore Passphrase.
    e. Ensure that the Confirm Custom Trust Keystore path is pointing to the *ORACLE_BASE*/products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks file.
    f. Leave the Custom Trust Keystore type blank.
    g. Change the Custom Trust Keystore Passphrase entry. This should be the same as the *keystorepassword* field described in the first bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."
    h. Re-enter the Custom Trust Keystore Passphrase.
    i. Click **Save**.
15. Choose the **SSL** tab.

a. Make sure that Identity and Trust Locations is set to **Keystores**.

b. Change the Private Key Alias to `CRMHOST3_fusion`.

c. Change the Private Key Passphrase to the *keypassword*, as described in the second bullet in Step 4 in Section 6.2, "Creating the Identity Keystore on CRMHOST2."

d. Re-enter the *keypassword* from Step c for the Confirm Private Key Passphrase.

e. Click **Save**.

16. Click **Activate Changes**.

17. Set the following variable:

```
WLST_PROPERTIES="-Dweblogic.security.SSL.trustedCAKeyStore=ORACLE_BASE/
products/fusionapps/wlserver_10.3/server/lib/fusion_trust.jks"
```

18. Restart the domain's Administration Server:

```
CRMHOST3> ORACLE_BASE/products/fusionapps/wlserver_10.3/common/bin/wlst.sh

CRMHOST3> nmConnect(username='username', password='password',
domainName='CRMDomain', host='CRMHOST1',port='5556',
nmType='ssl', domainDir='/u01/oracle/config/domains/CRMHOST1/CRMDomain')

CRMHOST3> nmStart('AdminServer')
```

Note: The *username* and *password* used in the `nmConnect` are the Node Manager credentials (username and password) specified when creating the provisioning plan. This is shown in Figure 4–3 in "Using the Provisioning Process to Install Components for an Enterprise Deployment".

---

19. Log in to the Administration Server once again (`http://crminternal.mycompany.com:7777/console`) and verify that all the managed servers, including scaled-out servers, are running.

Note: For all the scaled-up and scaled-out servers, change the Arguments in the `/u02/local/oracle/config/domains/HOSTNAME/DomainName/servers/ManagedServer/data/nodemanager/startup.properties` file to the following:

```
Arguments=-DJDBCProgramName\=DS/CRMDomain/ManagedServer_2
-Dserver.group\=HelpPortalCluster
```

Note: For all the scaled-up and scaled-out managed servers, do the following:

1. Access the Oracle WebLogic Server Administration Console for the CRMDomain:

   `http://crminternal.mycompany.com:7777/console`

2. Navigate to **Environment > Servers** and click the "Managed Server" link.
3. First select the **Logging** tab and then the **HTTP** tab.
4. Update the following parameters:

   – Log file name: logs/access.log.%yyyyMMdd%

   – Rotation Type: By Time

   – Limit number of retained files: leave this option unchecked

   – Rotate log file on startup: leave this option unchecked

5. Click **Save**.
6. Expand **Advanced Node** and set the following:

   – Format: Extended

   – Extended Logging Format Fields:

   ```
   date time time-taken cs-method cs-uri sc-status
   sc(X-ORACLE-DMS-ECID) cs(ECID-Context) cs(Proxy-Remote-User)
   cs(Proxy-Client-IP)
   ```

7. Click **Save** and **Activate Changes**.
8. Restart the Managed Server for the changes to take affect.

18.1.3.5 Validating the System

You should verify URLs to ensure that the appropriate routing and failover is working from Oracle HTTP Server to the SalesCluster.

To verify the URLs:

1. Log in to the `CRMDomain` Oracle WebLogic Server Administration Console and stop all the managed servers on the *`CRMHOST1`* while the managed servers on *`CRMHOST3`* are running.
2. Access the following URLs to verify that routing and failover are functioning properly. (Ensure the log in prompt is visible.)
   - `http://CRMHOST1:9024/soa-infra`
   - `http://crminternal.mycompany.com:7777/soa-infra`
3. Log in to the `CRMDomain` Oracle WebLogic Server Administration Console and stop all the managed servers on *`CRMHOST3`*.
4. Start the managed servers on *`CRMHOST1`*.
5. Repeat Step 2. (Ensure the log in prompt is visible.)
6. Start all the managed servers on *`CRMHOST3`* and verify that they are running on *`CRMHOST1`*.

18.1.3.6 Additional Configuration Procedures for Scaling Out Oracle SOA Suite Server

At this point, `soa_server1` and `soa_server3` are running on *`CRMHOST1`* and *`CRMHOST3`*.

> **Note:** For Oracle Fusion Customer Relationship Management, the Oracle SOA Suite virtual IPs for *`CRMHOST1`* and *`CRMHOST3`* are called *`CRMSOAVH1`* and *`CRMSOAVH3`*.

This section includes the following topics:

- Enabling Virtual IPs on CRMHOST3
- Setting the Listen Address for soa_server3
- Configuring JMS for the Oracle SOA Suite Server
- Configuring Oracle Coherence for Deploying Composites
- Disabling Host Name Verification for the soa_servern Managed Servers
- Restarting Node Manager on CRMHOST3
- Starting and Validating soa_server3 on CRMHOST3

18.1.3.6.1 Enabling Virtual IPs on CRMHOST3 Do the following to enable the virtual IPs on Linux:

> **Note:** In this example `ethX` is the ethernet interface (`eth0` or `eth1`) and `Y` is the index (0, 1, 2, and so on).

1. Run the `ifconfig` command as root:

   ```
   /sbin/ifconfig interface:index IPAddress netmask netmask
   ```

   For example:

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

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

   ```
   /sbin/arping -q -U -c 3 -I interface IPAddress
   ```

   For example:

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

3. Validate that the address is available by pinging it from another node.

   For example:

   ```
   /bin/ping 100.200.140.206
   ```

18.1.3.6.2 Setting the Listen Address for soa_server3 Ensure that you have performed the steps described in Section 14.1 before setting the `soa_server3` listen address.

Perform these 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 **soa_server3** in the column of the table. The Settings page for `soa_server3` is displayed.
6. Set the **Listen Address** to *`CRMSOAVH3`*.
7. Click **Save**.
8. Click **Activate Changes**.
9. The changes will not take effect until the `soa_server3` Managed Server is restarted (ensure that Node Manager is up and running):
   a. On the Summary of Servers page, select the **Control** tab.
   b. Select **soa_server3** in the table and then click **Shutdown**.
   c. After the server has shut down, select **soa_server3** in the table and then click **Start**.

18.1.3.6.3 Updating the FusionVirtualHost_crm.conf Configuration File For information, see Section 14.4, "Updating the FusionVirtualHost_crm.conf Configuration File."

18.1.3.6.4 Configuring JMS for the Oracle SOA Suite Server After *`CRMHOST1`* has been provisioned, the JMS server and file store are set up and configured for *`CRMHOST1`*. You now must configure the file store for *`CRMHOST3`*. Configure the location for all persistence stores to a directory visible from both nodes.

To configure the file store for *`CRMHOST3`*:

1. Log in to the Oracle WebLogic ServerAdministration Console.
2. In the Domain Structure window, expand the **Services** node and then click the **Persistence Stores** node.

   The Summary of Persistence Stores page appears.
3. Click **Lock & Edit**.
4. Click **New**, and then **Create File Store**.
5. Enter a name (for example, `SOAJMSFileStore_auto_3`), and a target, `soa_server3`:

   `ORACLE_BASE``/config/domains/`*`CRMHOST1`*`/CRMDomain`
6. Click **OK** and activate the changes.
7. In the Domain Structure window, expand the **Services** node and then click the **Messaging > JMS Servers** node.

   The Summary of JMS Servers page appears.
8. Click **Lock & Edit**.
9. Click **New**.
10. Enter a name (for example, `SOAJMSServer_3`), then select **SOAJMSFileStore_auto_3** in the Persistence Store dropdown list.
11. Click **Next**.
12. Select **soa_server3** as the target.

13. Click **Finish** and **Activate Changes**.
14. In the Domain Structure window, expand the **Services** node and then click the **Messaging > JMS Modules** node.

    The JMS Modules page appears.
15. In the Change Center, click **Lock & Edit**.
16. Click **SOAJMSModule** and then click the **Subdeployments** tab.
17. Select **SOAJMSServer** under **Subdeployments**.
18. Add the new `SOAJMSServer_3` as additional targets for the subdeployment.
19. Click **Save** and **Activate Changes**.

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

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

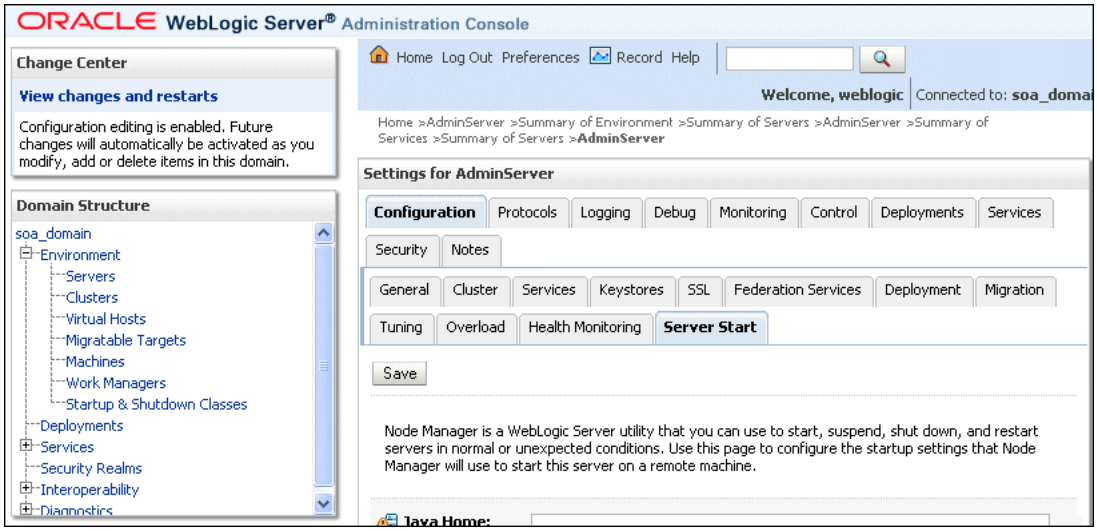
Multicast communication enables Oracle Fusion Middleware SOA to discover all of the members of a cluster to which it deploys composites dynamically. However, unicast communication does not enable nodes to discover other cluster members in this way. Consequently, you must specify the nodes that belong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

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

Specify the nodes using the `tangosol.coherence.wka` *n* system property, where *n* is the number for each Oracle HTTP Server. The numbering starts at 1. This numbering must be sequential and must not contain gaps. In addition, specify the host name used by Oracle Coherence to create a cluster through the `tangosol.coherence.localhost` system property. This local host name should be the virtual host name used by the SOA server as the listener addresses. Set this property by adding the `-Dtangosol.coherence.localhost` parameters to the Arguments field of the Oracle WebLogic Server Administration Console's Server Start tab, shown in Figure 18–1.

Note: *CRMSOAVH1* is the virtual host name that maps to the virtual IP where `soa_server1` is listening (in *CRMHOST1*). *CRMSOAVH3* is the virtual host name that maps to the virtual IP where `soa_server3` is listening (in *CRMHOST3*).

Figure 18–1 Setting the Host Name



To add the host name used by Oracle Coherence:

1. Log in to the Oracle WebLogic Server Administration Console.
2. In the Domain Structure window, expand the **Environment** node.
3. Click **Servers**.

   The Summary of Servers page appears.
4. Select **soa_server1** (represented as a hyperlink) from the column of the table.

   The Settings page appears.
5. Click **Lock & Edit**.
6. Click the **Server Start** tab (shown in Figure 18–1).
7. Enter the following for `soa_server1` and `soa_server3` into the **Arguments** field.

   For `soa_server1`, enter the following:

```
-Dtangosol.coherence.wka1=CRMSOAVH1
-Dtangosol.coherence.wka2=CRMSOAVH3
-Dtangosol.coherence.localhost=CRMSOAVH1
-Dtangosol.coherence.localport=8089
-Dtangosol.coherence.wka1.port=8089
-Dtangosol.coherence.wka2.port=8089
```

   For `soa_server3`, enter the following:

```
-Dtangosol.coherence.wka1=CRMSOAVH3
-Dtangosol.coherence.wka2=CRMSOAVH1
-Dtangosol.coherence.localhost=CRMSOAVH3
-Dtangosol.coherence.localport=8089
-Dtangosol.coherence.wka1.port=8089
```

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

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

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

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

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

Note: The Oracle Coherence cluster used for deployment uses port 8088 by default. This port can be changed by specifying the `-Dtangosol.coherence.wkaX.port` startup parameter.

18.1.3.6.6 Disabling Host Name Verification for the soa_server*n* Managed Servers

This step is required if you have not set up the appropriate certificates to authenticate the different nodes with the Administration Server. By default, Host Name Verification should be set to `None`. If it is not, follow the steps below.

If you have not configured the server certificates, you will receive errors when managing the different WebLogic servers. To avoid these errors, disable host name verification while setting up and validating the topology, and enable it again once the enterprise deployment topology configuration is complete.

To disable Host Name Verification:

1. Log in to Oracle WebLogic Server Administration Console. For example, `http://crminternal.mycompany.com:777/console`.
2. Click **Lock & Edit**.
3. Expand the **Environment** node in the Domain Structure window.
4. Click **Servers**.

   The Summary of Servers page appears.
5. Select **soa_server3** (represented as a hyperlink) from the column of the table.

   The Settings page appears.

6. Select the **SSL** tab.
7. Expand the **Advanced** section of the page.
8. Set **Hostname Verification** to **None**.
9. Click **Save** and activate the changes.

18.1.3.6.7 Restarting Node Manager on CRMHOST3 To restart Node Manager on *CRMHOST3*, follow the steps in Section 14.9, "Restarting Node Manager on CRMHOST1."

18.1.3.6.8 Starting and Validating soa_server3 on CRMHOST3 To start the `soa_server3` managed server on *CRMHOST3* and ensure that it is configured correctly:

1. From the Administration Console, start the `soa_server3` managed server.
2. Verify that the server status is reported as "Running" in the Administration Console. If the server is shown as "Starting" or "Resuming," wait for the server status to change to "Started." If another status is reported (such as "Admin" or "Failed"), check the server output log files for errors.
3. Access `http://`*`CRMSOAVH3`*`:9024/soa-infra` and `http://`*`crminternal`*`.mycompany.com:7777/soa-infra`.

18.1.4 Scaling Up the Topology (Adding Managed Servers to an Existing Node) for Oracle SOA Suite Server

Before performing the procedures in this section, ensure that `CommonDomain` and its managed servers are running.

18.1.4.1 Cloning Managed Servers and Assigning Them to CRMHOST3

To add a managed server and assign it to *CRMHOST3*:

1. Log in to the Administration Server: `http://`*`crminternal`*`.mycompany.com:7777/console`.
2. Navigate to **CRMDomain > Environment > Servers**.
3. Switch to **Lock & Edit** mode.
4. Select the ***Managed_Servers*** checkbox (for example, **soa_server3**) and then click **Clone**.
5. Specify the following Server Identity attributes:
   - Server Name - `soa_server4`

     Note: To ensure consistency in naming, copy the name of the server shown in **Server Identity** and paste it into the **Server Name** field. Then change the number to "_4".

   - Server Listen Address - <*CRMHOST3*>
   - Server Listen Port - leave "Give an unused port on the machine *CRMHOST3*"
6. Click **OK**.
7. Navigate back to **CRMDomain > Environment > Servers**. You now should see the newly cloned SOA server, `soa_server4`.

8. Click **soa_server4** and change the following attributes:
   - Machine - <*CRMHOST3*>
   - Cluster Name - Default, SOACluster
9. From **soa_server4**, click **Advanced** and then select the **WebLogic Plug-In Enabled** checkbox.
10. Run the newly created managed server:
    a. Navigate to **CRMDomain > Environment**.
    b. From the **Navigation** pane on the Oracle WebLogic Server console, select **Activate Changes**.
    c. Navigate to **CRMDomain > Environment > Servers > Control**.
    d. Check the newly created managed server and click **Start**.
    e. Navigate to **CRMDomain > Environment > Servers** and check the **State** to verify that the newly created managed servers are running.
11. Log in to the Administration Server once again (`http://`*`crminternal`*`.mycompany.com:7777/console`) and verify that all the managed servers, including scaled-up servers, are running.

18.1.4.2 Validating the System

You should verify URLs to ensure that the appropriate routing and failover is working from Oracle HTTP Server to the SalesCluster.

To verify the URLs:

1. Log in to the `CRMDomain` Oracle WebLogic Server Administration Console and stop the `soa_server1`, `soa_server2`, and `soa_server3` scaled-up managed servers on *`CRMHOST1`*, *`CRMHOST2`*, and *`CRMHOST3`*.
2. Access the following URLs to verify that routing and failover are functioning properly. (Ensure the log in prompt is visible.)
   - `http://CRMHOST1:9024/soa-infra`
   - `http://crminternal.mycompany.com:7777/soa-infra`
3. Log in to the `CRMDomain` Oracle WebLogic Server Administration Console and stop all the `soa_server4` managed servers on *`CRMHOST3`*.
4. Start the managed servers on *`CRMHOST1`*.
5. Repeat Step 2. (Ensure the log in prompt is visible.)
6. Start all the managed servers on *`CRMHOST2`* and verify that they are running on *`CRMHOST1`*.

18.1.5 Scaling Out the Topology (Adding Managed Servers to a New Node) for Oracle Business Intelligence

When scaling out the topology, you add a new Managed Server and set of system components to a new node in your topology (CRMHOST3). 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.

18.1.5.1 Prerequisites for Scaling Out the Topology for Oracle Business Intelligence

Before performing the steps in this section, ensure that you meet these requirements:

- There must be existing nodes running OOracle Business Intelligence Managed Servers within the topology.
- The new node (*CRMHOST3*) can access the existing home directories for Oracle WebLogic Server and Oracle Business Intelligence.
- When an *FA_MW_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_BASE*`/products/fusionapps/bi/oui/bin/attachHome.sh`. To update the Middleware home list to add or remove a *WL_HOME*, edit the *ORACLE_BASE*`/products/fusionapps/.home file`. See the steps below.
- You must ensure that all shared storage directories are available on the new node. Ensure that all shared directories are available on all nodes, except for the *ORACLE_INSTANCE* directory and the domain directory for the scaled-out Managed Server.

  Also, if you are using shared storage for the identity keystore and trust keystore that hold your host name verification certificates, make sure that the shared storage directory is accessible from the scaled-out node (*CRMHOST3*). If you are using local directories for your keystores, follow the steps in Section 6.2, "Creating the Identity Keystore on CRMHOST2" 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*

18.1.5.2 Scale-Out Procedure for Oracle Business Intelligence

Perform these steps to scale out Oracle Business Intelligence on *CRMHOST3*:

1. On *CRMHOST3*, mount the existing Middleware home, which should include 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, just like the rest of the nodes in the domain.
2. Run the following command to attach *ORACLE_BASE*`/products/fusionapps/oracle_common` in shared storage to the local Oracle Inventory:

```
CRMHOST3> cd ORACLE_BASE/products/fusionapps/oracle_common/oui/bin/
CRMHOST3> ./attachHome.sh -jreLoc ORACLE_BASE/products/fusionapps/jdk6
```

To update the Middleware home list, create (or edit, if another WebLogic installation exists in the node) the *ORACLE_BASE*/products/fusionapps/.home file and add *ORACLE_BASE*/products/fusionapps to it.

3. Start Node Manager:
   a. Stop any Node Manager running on *CRMHOST3*.
   b. Change directory to *ORACLE_BASE*/products/fusionapps/wlserver_10.3/common/nodemanager and edit the nodemanager.properties file with the following:

      ```
      SecureListener=false
      ```

   c. Change directory to *ORACLE_BASE*/products/fusionapps/oracle_common/common/bin and run the following script:

      ```
      ./setNMProps.sh
      ```

   d. Change directory to *ORACLE_BASE*/products/fusionapps/wlserver_10.3/server/bin and run the following script:

      ```
      ./startNodeManager.sh
      ```

      Node Manager starts on *CRMHOST3*.

      Note: Steps b through d will enable Node Manager on *CRMHOST3* and the Administrator Console to communicate on Plain Socket.

4. Run the Configuration Assistant from one of the shared Oracle homes, using the steps in Section 13.5.1, "Scaling Out the BI System on CRMHOST2" as a guide.
5. Scale out the system components on *CRMHOST3*, using the steps in Section 13.5.3, "Scaling Out the System Components" as a guide.
6. Configure the bi_server3 Managed Server by setting the Listen Address and disabling host name verification, using the steps in Section 13.5.5, "Configuring the bi_server2 Managed Server" as a guide.
7. Configure JMS for Oracle BI Publisher, as described in Section 13.5.6.3.3, "Configuring JMS for BI Publisher."
8. Configure Oracle BI for Microsoft Office on *CRMHOST3*, as described in Section 13.5.6.4, "Additional Configuration Tasks for Oracle BI for Microsoft Office."
9. Set the location of the default persistence store for bi_server3, as described in Section 13.5.7, "Configuring a Default Persistence Store for Transaction Recovery."
10. Configure Oracle HTTP Server for *BIVH3* using the steps in Section 13.4.3, "Updating the FusionVirtualHost_bi.conf Configuration File" as a guide.
11. Start the bi_server3 Managed Server and the system components running on *CRMHOST3*. See Section 13.5.8, "Starting and Validating Oracle Business Intelligence on CRMHOST2"for details.
12. Set up server migration for the new node, as described in the following sections:
    - Section 6.2, "Creating the Identity Keystore on CRMHOST2"
    - Section 16.2.4, "Editing Node Manager's Properties File"

- Section 16.2.5, "Setting Environment and Superuser Privileges for the wlsifconfig.sh Script"
- Section 16.2.6, "Configuring Server Migration Targets"
- Section 16.2.7, "Testing the Server Migration"

13. Access the following URLS to validate the configuration:
    - `http://BIVH3:10217/analytics` to verify the status of `bi_server3`.
    - `http://BIVH3:10217/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.

    - `http://BIVH3:10217/xmlpserver` to verify the status of the Oracle BI Publisher application.
    - `http://BIVH3:10217/ui` to verify the status of the Oracle Real-Time Decisions application.
    - `http://BIVH3:10217:/mapviewer` to verify the status of the map view functionality in Oracle BI EE.
    - `http://BIVH3:10217/hr` to verify Financial Reporting.
    - `http://BIVH3:10217/calcmgr/index.htm` to verify Calculation Manager.
    - `http://BIVH3:10217/aps/Test` to verify APS.
    - `http://BIVH3:10217/workspace` to verify workspace

Oracle recommends using host name verification for the communication between Node Manager and the servers in the domain. This requires the use of certificates for the different addresses communicating with the Administration Server and other servers. See Chapter 6, "Configuring Node Manager" for further details.

18.1.6 Scaling Up the Topology for Oracle Business Intelligence

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 running on one of your existing nodes.

Note that it is not necessary to run multiple Managed Servers on a given node.

18.1.6.1 Scale-Up Procedure for Oracle Business Intelligence

Perform these steps to scale up Oracle Business Intelligence on *CRMHOST3*:

1. Log in to Fusion Middleware Control.
2. Expand the **Business Intelligence** node in the Farm_BIDomain window.
3. Click **coreapplication**.
4. Click **Capacity Management**, then click **Scalability**.

5. Change the number of BI Servers, Presentation Servers, or JavaHosts using the arrow keys.
6. Click **Apply**, then click **Activate Changes**.
7. Click **Restart** to apply recent changes.
8. Click **Restart** under **Manage System**.
9. Click **Yes** in the confirmation dialog.

18.2 Performing Backups and Recoveries

Table 18–1 lists the static artifacts to back up in the 11*g* Oracle Fusion Customer Relationship Management enterprise deployment.

Table 18–1 Static Artifacts to Back Up in the 11g CRM Enterprise Deployment

Type	Host	Location	Tier
ORACLE HOME (DB)	*FUSIONDBHOST1* and *FUSIONDBHOST2*	The location is user-defined. Generally, `/u01/oracle`.	Data tier
MW HOME (OHS)	*WEBHOST1* and *WEBHOST2*	*ORACLE_BASE*`/products` *ORACLE_BASE*`/config`	Web tier
MW HOME (/APPS_HOME)	*CRMHOST1* and *CRMHOST2*	*ORACLE_BASE*`/products` *ORACLE_BASE*`/config`	Application tier
Installation-related files		`/u01/oracle/repository`, OraInventory, `.home`, `oraInst.loc`, `oratab`	N/A

Table 18–2 lists the run-time artifacts to back up in the 11*g* Oracle Fusion Customer Relationship Management enterprise deployment.

Table 18–2 Run-Time Artifacts to Back Up in the 11g CRM Enterprise Deployment

Type	Host	Location	Tier
DOMAIN HOME	*CRMHOST1* and *CRMHOST2*	`/u02/local/oracle/config/domains/HOSTNAME/domain_name`	Application tier
Application artifacts (EAR and WAR files)	*CRMHOST1* and *CRMHOST2*	Find the application artifacts by viewing all of the deployments through the administration console.	Application tier
Oracle HTTP Server instance home	*WEBHOST1* and *WEBHOST2*	*ORACLE_BASE*`/config/CommonDomain_webtier`	Web tier
Oracle RAC databases	*FUSIONDBHOST1* and *FUSIONDBHOST2*	The location is user-defined.	Data tier

For more information on backup and recovery of Oracle Fusion Middleware components, see *Oracle Fusion Middleware Administrator's Guide*.

18.3 Monitoring the Topology

For information on monitoring the Oracle Fusion Customer Relationship Management topology, see the following documents:

- *Oracle Fusion Middleware System Administrator's Guide for Content Server*
- *Oracle Fusion Middleware Application Administrator's Guide for Content Server*
- *Oracle Fusion Middleware Administrator's Guide for Imaging and Process Management*

18.4 Migrating from a Test Environment to a Production Environment

For information, see "Moving Components for Oracle Fusion Applications Across Environments" in the *Oracle Fusion Applications Administrator's Guide*.

18.5 Configuring Log File Rotation

An **ODL log** is a set of log files that includes the current ODL log file and zero or more **ODL Archives (segment files)** that contain older messages. As the log file grows, new information is added to the end of the log file, *server_name*-diagnostic.log. When the log file reaches the rotation point, it is renamed and a new log file, *server_name*-diagnostic.log is created. You specify the rotation point, by specifying the maximum ODL segment size, or, for the log files of some components, the rotation time and rotation frequency.

Segment files are created when the ODL log file *server_name*-diagnostic.log reaches the rotation point. That is, the *server_name*-diagnostic.log is renamed to *server_name*-diagnostic-*n*.log, where *n* is an integer, and a new *server_name*-diagnostic.log file is created when the component generates new diagnostic messages.

To limit the size of the ODL log, you can specify:

- The maximum size of the logging directory. Whenever the sum of the sizes of all of the files in the directory reaches the maximum, the oldest archive is deleted to keep the total size under the specified limit.

  By default, the log files are rotated when they reach 10 MB. The maximum size of all log files for a particular component is 100 MB.

- The maximum size of the log file. You specify that a new log file be created when a specific time or frequency is reached.

> **Note:** After you change the log file rotation, the configuration is reloaded dynamically. It may take 1 or 2 seconds to reload the configuration.

The following topics describe how to change the rotation:

- Section 18.5.1, "Specifying Log File Rotation Using Fusion Middleware Control"
- Section 18.5.2, "Specifying Log File Rotation Using WLST"

18.5.1 Specifying Log File Rotation Using Fusion Middleware Control

To configure log file rotation using Fusion Middleware Control for a component:

1. From the navigation pane, select the component.
2. From the dynamic target menu, choose **Logs,** then **Log Configuration.**

   The Log Configuration page is displayed.
3. Select the Log Files tab.

4. In the table, select the logger and click **Edit Configuration.**

   The Edit Log File dialog box is displayed.

5. In the Rotation Policy section, you can select one of the following:
   - **Size Based:** If you select this, enter the following:
     - For **Maximum Log File Size,** enter the size in MB, for example, 15.
     - For **Maximum Size of All Log Files,** enter the size in MB, for example, 150.
   - **Time Based:** If you select this, enter the following:
     - For **Start Time,** enter the date when you want the rotation to start. For example, enter 10-SEP-2009.
     - For **Frequency,** you can select **Minutes** and enter the number of minutes, or you can select **Hourly, Daily,** or **Weekly.**
     - For **Retention Period,** you can specify how long the log files are kept. You can select **Minutes** and enter the number of minutes, or you can specify **Day, Week**, **Month,** or **Year.**

       Specifying a shorter period means that you use less disk space, but are not able to retrieve older information.

6. Click **OK.**
7. In the confirmation window, click **Close.**

18.5.2 Specifying Log File Rotation Using WLST

To specify log file rotation using WLST, use the `configureLogHandler` command. You can specify size-based rotation or time-based rotation.

For example, to specify that the log files rotate daily and that they are retained for a week, use the following command:

```
configureLogHandler(name='odl-handler', rotationFrequency='daily',
                    retentionPeriod='week')
```

To specify that the size of a log file does not exceed 5MB and rotates when it reaches that size, use the following command:

```
configureLogHandler(name='odl-handler', maxFileSize='5M')
```

18.6 Patching the Topology

For information, see the *Oracle Fusion Applications Patching Guide*.

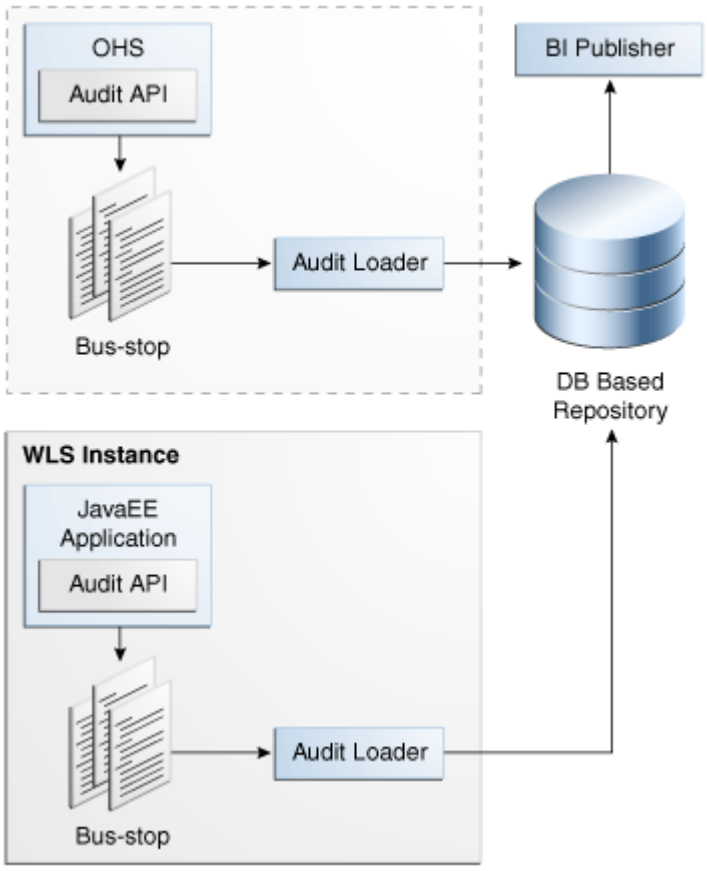
18.7 Auditing

Oracle Fusion Middleware Audit Framework is a new service in Oracle Fusion Middleware 11*g*, designed to provide a centralized audit framework for the middleware family of products. The framework provides audit service for platform components such as Oracle Platform Security Services (OPSS) and Oracle Web Services. It also provides a framework for JavaEE applications, starting with Oracle's own JavaEE components. JavaEE applications will be able to create application-specific audit events. For non-JavaEE Oracle components in the middleware, such as C or

JavaSE components, the audit framework also provides an end-to-end structure similar to that for JavaEE applications.

Figure 18–2 is a high-level architectural diagram of the Oracle Fusion Middleware Audit Framework.

Figure 18–2 Audit Event Flow



The Oracle Fusion Middleware Audit Framework consists of the following key components:

- **Audit APIs:** These are APIs provided by the audit framework for any audit-aware components integrating with the Oracle Fusion Middleware Audit Framework. During run time, applications may call these APIs, where appropriate, to audit the necessary information about a particular event happening in the application code. The interface allows applications to specify event details such as user name and other attributes needed to provide the context of the event being audited.
- **Audit Events and Configuration:** The Oracle Fusion Middleware Audit Framework provides a set of generic events for convenient mapping to application audit events. Some of these include common events such as authentication. The framework also allows applications to define application-specific events.

  These event definitions and configurations are implemented as part of the audit service in Oracle Platform Security Services. Configurations can be updated through Enterprise Manager (UI) and WebLogic Scripting Tool (WLST) command-line tool.

- **Audit Bus-stop:** Bus-stops are local files containing audit data before they are pushed to the audit repository. In the event where no database repository is configured, these bus-stop files can be used as a file-based audit repository. The bus-stop files are simple text files that can be queried easily to look up specific audit events. When a DB-based repository is in place, the bus-stop acts as an intermediary between the component and the audit repository. The local files are periodically uploaded to the audit repository based on a configurable time interval.
- **Audit Loader:** As the name implies, the audit loader loads the files from the audit bus-stop into the audit repository. In the case of platform and JavaEE application audit, the audit loader is started as part of the JavaEE container start-up. In the case of system components, the audit loader is a periodically spawned process.
- **Audit Repository:** The audit repository contains a predefined Oracle Fusion Middleware Audit Framework schema, created by Repository Creation Utility (RCU). Once configured, all the audit loaders are aware of the repository and upload data to it periodically. The audit data in the audit repository is expected to be cumulative and will grow over time. Ideally, this should not be an operational database used by any other applications; rather, it should be a standalone RDBMS used for audit purposes only. In a highly available configuration, Oracle recommends that you use an Oracle Real Application Clusters (RAC) database as the audit data store.
- **Oracle Business Intelligence Publisher:** The data in the audit repository is exposed through predefined reports in Oracle Business Intelligence Publisher. The reports allow users to drill down the audit data based on various criteria. For example:
  - User name
  - Time range
  - Application type
  - Execution context identifier (ECID)

For more introductory information for the Oracle Fusion Middleware Audit Framework, see the "Introduction to Oracle Fusion Middleware Audit Framework" chapter in the *Oracle Containers for J2EE Security Guide*.

For information on how to configure the repository for Oracle Fusion Middleware Audit Framework, see the "Configuring and Managing Auditing" chapter in the *Oracle Containers for J2EE Security Guide*.

The enterprise deployment topology does not include Oracle Fusion Middleware Audit Framework configuration. The ability to generate audit data to the bus-stop files and the configuration of the audit loader will be available once the products are installed. The main consideration is the audit database repository where the audit data is stored. Because of the volume and the historical nature of the audit data, it is strongly recommended that customers use a separate database from the operational store or stores being used for other middleware components.

18.8 Troubleshooting

This section covers the following topics:

- Section 18.8.1, "Page Not Found When Accessing soa-infra Application Through Load Balancer"

- Section 18.8.2, "Soa-infra Application Fails to Start Due to Deployment Framework Issues (Coherence)"
- Section 18.8.3, "Incomplete Policy Migration After Failed Restart of SOA Server"
- Section 18.8.4, "Oracle SOA Suite Server Fails to Start Due to Maximum Number of Processes Available in Database"
- Section 18.8.5, "Administration Server Fails to Start After a Manual Failover"
- Section 18.8.6, "Error While Activating Changes in Administration Console"
- Section 18.8.7, "SOA Server Not Failed Over After Server Migration"
- Section 18.8.8, "SOA Server Not Reachable From Browser After Server Migration"
- Section 18.8.9, "Oracle Access Manager Configuration Tool Does Not Remove URLs"
- Section 18.8.10, "Redirecting of Users to Login Screen After Activating Changes in Administration Console"
- Section 18.8.11, "Redirecting of Users to Administration Console's Home Page After Activating Changes to OAM"
- Section 18.8.12, "Configured JOC Port Already in Use"
- Section 18.8.13, "Out-of-Memory Issues on Managed Servers"
- Section 18.8.14, "JDBC Connection Reset Appears When on OEL 5.4"
- Section 18.8.15, "Missing JMS Instances on Oracle BI Publisher Scheduler Diagnostics Page"
- Section 18.8.16, "Oracle BI Publisher Jobs in Inconsistent State After Managed Server Shutdown"
- Section 18.8.18, "Administration Console Redirects from Internal URL to Container URL after Activation"

18.8.1 Page Not Found When Accessing soa-infra Application Through Load Balancer

Problem: A 404 "page not found" message is displayed in the web browser when you try to access the soa-infra application using the load balancer address. The error is intermittent and Oracle SOA Suite servers appear as "Running" in the WLS Administration Console.

Solution: Even when the Oracle SOA Suite managed servers may be up and running, some of the applications contained in them may be in Admin, Prepared or other states different from Active. The soa-infra application may be unavailable while the Oracle SOA Suite server is running. Check the Deployments page in the Administration Console to verify the status of the soa-infra application. It should be in "Active" state. Check the Oracle SOA Suite server's output log for errors pertaining to the soa-infra application and try to start it from the Deployments page in the Administration Console.

18.8.2 Soa-infra Application Fails to Start Due to Deployment Framework Issues (Coherence)

Problem: The soa-infra application fails to start after changes to the Coherence configuration for deployment have been applied. The Oracle SOA Suite server output log reports the following:

```
Cluster communication initialization failed. If you are using multicast, Please
make sure multicast is enabled on your network and that there is no interference
on the address in use. Please see the documentation for more details.
```

Solutions:

- When using multicast instead of unicast for cluster deployments of Oracle SOA Suite composites, a message similar to the above may appear if a multicast conflict arises when starting the soa-infra application (that is, starting the managed server on which Oracle SOA Suite runs). These messages, which occur when Oracle Coherence throws a run-time exception, also include the details of the exception itself. If such a message appears, check the multicast configuration in your network. Verify that you can ping multicast addresses. In addition, check for other clusters that may have the same multicast address but have a different cluster name in your network, as this may cause a conflict that prevents soa-infra from starting. If multicast is not enabled in your network, you can change the deployment framework to use unicast as described in *Oracle Coherence Developer's Guide*.
- When entering well-known address list for unicast (in server start parameters), make sure that the node's addresses entered for the localhost and clustered servers are correct. Error messages like the following are reported in the server's output log if any of the addresses is not resolved correctly:

  ```
  oracle.integration.platform.blocks.deploy.CompositeDeploymentCoordinatorMessage
  s errorUnableToStartCoherence
  ```

18.8.3 Incomplete Policy Migration After Failed Restart of SOA Server

Problem: The SOA server fails to start through the administration console *before* setting Node Manager property `startScriptEnabled=true`. The server does not come up after the property is set either. The SOA Server output log reports the following:

```
SEVERE: <.> Unable to Encrypt data
Unable to Encrypt data.
Check installation/post-installation steps for errors. Check for errors during SOA
server startup.

ORABPEL-35010
 .
Unable to Encrypt data.
Unable to Encrypt data.
Check installation/post-installation steps for errors. Check for errors
 during SOA server startup.
 .
 at
oracle.bpel.services.common.util.EncryptionService.encrypt(EncryptionService.java:
56)
...
```

Solution: Incomplete policy migration results from an unsuccessful start of the first SOA server in a cluster. To enable full migration, edit the `<jazn-policy>` element the system-jazn-data.xml file to grant permission to bpm-services.jar:

```
<grant>
  <grantee>
    <codesource>
<url>file:${oracle.home}/soa/modules/oracle.soa.workflow_11.1.1/bpm-services.jar
</url>
    </codesource>
  </grantee>
  <permissions>
    <permission>
      <class>java.security.AllPermission</class>
    </permission>
  </permissions>
</grant>
```

18.8.4 Oracle SOA Suite Server Fails to Start Due to Maximum Number of Processes Available in Database

Problem: A Oracle SOA Suite server fails to start. The domain has been extended for new types of managed server or the system has been scaled up (added new servers of the same type). The Oracle SOA Suite server output log reports the following:

```
<Warning> <JDBC> <BEA-001129> <Received exception while creating connection for
pool "SOADataSource-rac0": Listener refused the connection with the following
error:

ORA-12516, TNS:listener could not find available handler with matching protocol
stack >
```

Solution: Verify the number of processes in the database and adjust accordingly. As the SYS user, issue the `SHOW PARAMETER` command:

```
SQL> SHOW PARAMETER processes
```

Set the initialization parameter using the following command:

```
SQL> ALTER SYSTEM SET processes=greater than 2500 SCOPE=SPFILE
```

Restart the database.

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

18.8.5 Administration Server Fails to Start After a Manual Failover

Problem: Administration Server fails to start after the Administration Server node failed and manual failover to another nodes is performed. The Administration Server output log reports the following:

```
<Feb 19, 2009 3:43:05 AM PST> <Warning> <EmbeddedLDAP> <BEA-171520> <Could not
obtain an exclusive lock for directory: ORACLE_BASE/admin/edg_domain/aserver/edg_
domain/servers/AdminServer/data/ldap/ldapfiles. Waiting for 10 seconds and then
retrying in case existing WebLogic Server is still shutting down.>
```

Solution: When restoring a node after a node crash and using shared storage for the domain directory, you may see this error in the log for the Administration Server due to unsuccessful lock cleanup. To resolve this error, remove the file *ORACLE_HOME*/config/domains/CRMHOST1/DomainName/servers/AdminServer/data/ldap/ldapfiles/EmbeddedLDAP.lok.

18.8.6 Error While Activating Changes in Administration Console

Problem: Activation of changes in Administration Console fails after changes to a server's start configuration have been performed. The Administration Console reports the following when clicking **Activate Changes**:

```
An error occurred during activation of changes, please see the log for details.
 [Management:141190]The commit phase of the configuration update failed with an
exception:
In production mode, it's not allowed to set a clear text value to the property:
PasswordEncrypted of ServerStartMBean
```

Solution: This may happen when start parameters are changed for a server in the Administration Console. In this case, provide user name/password information in the server start configuration in the Administration Console for the specific server whose configuration was being changed.

18.8.7 SOA Server Not Failed Over After Server Migration

Problem: After reaching the maximum restart attempts by local Node Manager, Node Manager in the failover node tries to restart it, but the server does not come up. The server seems to be failed over as reported by Node Manager's output information. The VIP used by the SOA server is not enabled in the failover node after Node Manager tries to migrate it (if config in the failover node does not report the VIP in any interface). Executing the command "sudo ifconfig $INTERFACE $ADDRESS $NETMASK" does not enable the IP in the failover node.

Solution: The rights and configuration for `sudo` execution should not prompt for a password. Verify the configuration of `sudo` with your system administrator so that `sudo` works without a password prompt.

18.8.8 SOA Server Not Reachable From Browser After Server Migration

Problem: Server migration is working (SOA server is restarted in the failed over node), but the `Virtual_Hostname:8001/soa-infra` URL cannot be accessed in the web browser. The server has been "killed" in its original host and Node Manager in the failover node reports that the VIP has been migrated and the server started. The VIP used by the SOA server cannot be pinged from the client's node (that is, the node where the browser is being used).

Solution: The `arping` command executed by Node 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.

18.8.9 Oracle Access Manager Configuration Tool Does Not Remove URLs

Problem: The Oracle Access Manager 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 Oracle Access Manager Configuration Tool again with the correct URLs completes successfully; however, when accessing Policy Manager, the incorrect URL is still there.

Solution: The Oracle Access Manager Configuration Tool only adds new URLs to existing policies when executed with the same `app_domain` name. To remove a URL, use the Policy Manager Console in Oracle Access Manager. Log on to the Access Administration site for Oracle Access Manager, click on My Policy Domains, click on the created policy domain (SOA_EDG), then on the Resources tab, and remove the incorrect URLs.

18.8.10 Redirecting of Users to Login Screen After Activating Changes in Administration Console

Problem: After configuring Oracle HTTP Server and LBR to access the Oracle WebLogic Administration Console, some activation changes cause the redirection to the login screen for the Administration Console.

Solution: This is the result of the console attempting to follow changes to port, channel, and security settings as a user makes these changes. For certain changes, the console may redirect to the Administration Server's listen address. Activation is completed regardless of the redirection. It is not required to log in again; users can simply update the URL to `crm.mycompany.com/console/console.portal` and directly access the home page for the Administration Console.

> **Note:** This problem will not occur if you have disabled tracking of the changes described in this section.

18.8.11 Redirecting of Users to Administration Console's Home Page After Activating Changes to OAM

Problem: After configuring OAM, some activation changes cause the redirection to the Administration Console's home page (instead of the context menu where the activation was performed).

Solution: This is expected when OAM SSO is configured and the Administration Console is set to follow configuration changes (redirections are performed by the Administration Server when activating some changes). Activations should complete regardless of this redirection. For successive changes not to redirect, access the Administration Console, choose Preferences, then Shared Preferences, and unselect the "Follow Configuration Changes" check box.

18.8.12 Configured JOC Port Already in Use

Problem: Attempts to start a managed server that uses the Java Object Cache (JOC), such as OWSM managed servers, fail. The following errors appear in the logs:

```
J2EE JOC-058 distributed cache initialization failure
J2EE JOC-043 base exception:
J2EE JOC-803 unexpected EOF during read.
```

Solution: Another process is using the same port that JOC is attempting to obtain. Either stop that process, or reconfigure JOC for this cluster to use another port in the recommended port range.

18.8.13 Out-of-Memory Issues on Managed Servers

Problem: You are experiencing out-of-memory issues on managed servers.

Solution: Increase the size of the memory heap allocated for the Java VM to at least one gigabyte:

1. Log in to the Oracle WebLogic Administration Console.
2. Click **Environment**, then **Servers**.
3. Click on a managed server name.
4. Open the **Configuration** tab.
5. Open the **Server Start** tab in the second row of tabs.
6. Include the memory parameters in the **Arguments** box, for example:

```
-Xms3072m -Xmx1024m -XX:CompileThreshold=8000 -XX:PermSize=128m
-XX:MaxPermSize=1024m
```

> **Note:** Please note that the memory parameter requirements may differ between various JVMs (Sun, JRockit, or others). See "Increasing the Java VM Heap Size for Managed Servers" in the *Oracle Fusion Middleware Installation Guide for Oracle Enterprise Content Management Suite* for further details.

7. Save the configuration changes.
8. Restart all running managed servers.

18.8.14 JDBC Connection Reset Appears When on OEL 5.4

Problem: When you are on Oracle Enterprise Linux (OEL) 5.4, a Java Database Connectivity (JBDC) connection reset appears.

Solutions:

- Upgrade to OEL 5.6.
- As root, do the following:
  1. Download and install the `rngd` tool.
  2. Execute the following commands (in order):

```
rngd -r /dev/urandom -o /dev/random

cat /proc/sys/kernel/random/entropy_avail
```

  3. Ensure that entropy returns a number greater than 1000.

18.8.15 Missing JMS Instances on Oracle BI Publisher Scheduler Diagnostics Page

In some cases, only one JMS instance is visible on the Oracle BI Publisher Scheduler diagnostics page, rather than all instances in the cluster. This issue is most likely caused by clocks being out of sync. For more information on the importance of

synchronizing clocks on all nodes in the cluster, see "Clock Synchronization" in the chapter "Database and Environment Preconfiguration" in *Oracle Fusion Middleware Enterprise Deployment Guide for Oracle Business Intelligence*.

18.8.16 Oracle BI Publisher Jobs in Inconsistent State After Managed Server Shutdown

Before shutting down the Managed Server on which Oracle BI Publisher is running, it is a best practice (but not mandatory) to wait for all running Oracle 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.

18.8.17 JMS Instance Fails in an Oracle BI Publisher Cluster

On rare occasions, a JMS instance is missing from an Oracle BI Publisher Scheduler cluster. To resolve this issue, restart the Oracle BI Publisher application from the Oracle WebLogic Server Administration Console.

To restart Oracle BI Publisher:

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**.

18.8.18 Administration Console Redirects from Internal URL to Container URL after Activation

Log in to the Administration Console and do the following:

1. Click **Preferences** in the top navigation bar.
2. Select the **Shared Preferences** tab.
3. De-select **Follow Configuration Changes**.
4. Click **Save** and then **Activate Changes**.

A

Protected Domain URIs

This appendix provides a list of URIs that need to be protected for each domain.

This appendix includes the following topics:

- Section A.1, "Protected CRM Domain URIs"
- Section A.2, "Protected BI Domain URIs"

A.1 Protected CRM Domain URIs

The following is the list of protected Oracle Fusion Customer Relationship Management domain URIs:

- `/cdmFoundation/adfAuthentication`
- `/DateEffectivitySyncEssTest/adfAuthentication`
- `/DataQualityBatchEssTest/adfAuthentication`
- `/DataQualitySync/adfAuthentication`
- `/FoundationBulkImportEssTest/adfAuthentication`
- `/resourcemgr/adfAuthentication`
- `/EssAppNew/adfAuthentication`
- `/CommonComponentApps-SuperWeb-context-root/adfAuthentication`
- `/commonComponents-SuperWeb-context-root/adfAuthentication`
- `/WorkMgmtSubmitBatchAsgn/adfAuthentication`
- `/CommonComponentsEss/adfAuthentication`
- `/SalesPartiesEss/adfAuthentication`
- `/SalesPartiesEssTest/adfAuthentication`
- `/contractManagement-SuperWeb-context-root/adfAuthentication`
- `/ContractManagementEss-ContractsWrkAreaAssgnOwnerPublicEssUiTest-context-root/adfAuthentication`
- `/contractManagementEss-ContractsCoreStatusMgmtPublicEssUi-context-root/adfAuthentication`
- `/contractManagementEss-ContractsTermsLibraryPublicEssUi-context-root/adfAuthentication`
- `/workflow/ContractsCoreApprovalUI/adfAuthentication`

- /CrmAnalytics-AnalyticsWebapp-context-root/adfAuthentication
- /CustomerCenter-SuperWeb-context-root/adfAuthentication
- /customerDataHub/adfAuthentication
- /TestDispatcherWebApp/
- /crm-marketing/adfAuthentication
- /crm-leads/adfAuthentication
- /orderCapture/adfAuthentication
- /outlookEdition-OutlookEditionConnectorPublicUI-context-root/adfAuthentication
- /sales-SuperWeb-context-root/adfAuthentication
- /salesPredictionEngine-SuperWeb-context-root/adfAuthentication
- /SetTransform/adfAuthentication
- /territories/adfAuthentication
- /cdmFoundation
- /DateEffectivitySyncEssTest
- /DataQualityBatchEssTest
- /DataQualitySync
- /FoundationBulkImportEssTest
- /resourcemgr
- /EssAppNew
- /CommonComponentApps-SuperWeb-context-root
- /commonComponents-SuperWeb-context-root
- /WorkMgmtSubmitBatchAsgn
- /CommonComponentsEss
- /SalesPartiesEss
- /SalesPartiesEssTest
- /contractManagement-SuperWeb-context-root
- /ContractManagementEss-ContractsWrkAreaAssgnOwnerPublicEssUiTest-context-root
- /contractManagementEss-ContractsCoreStatusMgmtPublicEssUi-context-root
- /contractManagementEss-ContractsTermsLibraryPublicEssUi-context-root
- /workflow/ContractsCoreApprovalUI
- /CrmAnalytics-AnalyticsWebapp-context-root
- /CustomerCenter-SuperWeb-context-root
- /customerDataHub
- /TestDispatcherWebApp

- /crm-marketing
- /crm-leads
- /orderCapture
- /outlookEdition-OutlookEditionConnectorPublicUI-context-root
- /sales-SuperWeb-context-root
- /salesPredictionEngine-SuperWeb-context-root
- /SetTransform

A.2 Protected BI Domain URIs

The following is the list of protected Oracle Business Intelligence domain URIs:

- /analytics/saw.dll
- /hr
- /aps
- /workspace
- /calcmgr
- /xmlpserver
- /ui
- /bioffice