Oracle® Fusion Middleware

Integration Guide for Oracle Identity Management Suite

12*c* (12.2.1.3.0)
E96173-03
April 2023

Oracle Fusion Middleware Integration Guide for Oracle Identity Management Suite, 12*c* (12.2.1.3.0)

E96173-03

Primary Authors: Debapriya Datta, KC Francis, Priscilla Lee, Vinaye Misra

Contributors: John Boyer, Damien Carru, Andre Correa, Sidhartha Das, Fabienne Dorson, Yagnesh Gajjar, Daniel Gralewski, Stephen Grenholm, Manish Gulati, Lancer Guo, Tiexin Guo, Lakshmi Hariharan, Achyut Jagtap, Dan Joyce, Rakesh K, Kevin Kessler, Rajesh Kishore, Simon Kissane, Peter LaQuerre, Wei Jie Lee, Eric Locatelli, Harsh Maheshwari, Tim Melander, Rajesh Pakkath, Nitin Patel, Paulo Pereira, Mehul Poladia, Sanjay Rallapalli, Deepak Ramakrishnan, Loganathan Ramasamy, Rima Rana, Ajit Raskar, Pardha Reddy, Sanjay Sadarangani, Abhimanyu Seth, Kuldeep Shah, Pulkit Sharma, Daniel Shih, Semyon Shulman, Bhupinder Singh, Uppili Srinivasan, Dawn Tyler, Yogaraja Thyagarajan, Rohit Tiwari, Ken Vincent, Ning Wang, Norman Wang, Mark Wilcox, Michele Williams, Haisheng Yu, Amy Yue

Contents

D Configuring Oracle Virtual Directory for Integration with Oracle Access Management Access Manager

List of Figures

List of Tables

Preface

This guide describes how you can integrate certain components in the Oracle Identity Management suite to provide a broad range of solutions for application environment including: integration with LDAP repositories, identity and access management, advanced login and password security, and identity federation.

Audience

This document is intended for administrators who wish to integrate Oracle Identity Management components using a simple topology without high availability features.

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 that have purchased support 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 documentation set:

- *Administering Oracle Access Management*
- *Oracle® Fusion Middleware Administrator's Guide for Oracle Virtual Directory*
- *Oracle® Fusion Middleware Third-Party Application Server Guide for Oracle Identity and Access Management*

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.

Convention	Meaning
`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

This preface provides a summary of new features and updates to Oracle Identity Management suite integration.

New and Changed Features for 12c (12.2.1.3.0)

Oracle Identity Management doesn't support integration of Oracle Access Management and Oracle Identity Governance using the IDMConfigTool. It supports integration of Oracle Access Management 12c and Oracle Adaptive Access Manager 11g R2PS3.

Integration Scenarios Not Supported in Release 12.2.1.3.0

Oracle Identity Management doesn't support the following integrations in Release 12.2.1.3.0 and onwards:

- Integration with Oracle Mobile Security Suite. See Oracle Support article Oracle Mobile Security Suite Statement Of Direction (Doc ID 2112686.1).
- Oracle Access Management and Oracle Identity Governance using the IDMConfigTool.

  If you are upgrading from 11.1.2.3 to 12.2.1.3.0, you can integrate Oracle Access Management and Oracle Identity Governance using this procedure Integrating Access Manager and Oracle Identity Manager.

Integration Scenarios Supported in Release 12.2.1.3.0

Oracle Identity Management supports the following integration in Release 12.2.1.3.0 and onwards:

- Oracle Access Management 12c and Oracle Adaptive Access Manager 11g R2PS3

Part I
IdM Integration Topology

This part introduces the integration topologies supported by this document, and describes the tools used during integration.

This part contains the following chapter:

- Introduction to IdM Suite Components Integration

1

Introduction to IdM Suite Components Integration

This chapter explains integration concepts for the Oracle Identity Management suite. The chapter contains these topics:

- Prerequisites to Integrating Oracle Identity Management Suite Components
- Understanding Oracle Identity Management Integration Topologies
- Overview of Oracle Identity Management Components Used in the Integration
- IdM Integration Quick Links
- Common Access Manager and Oracle Identity Governance Integration Scenarios
- System Requirements and Certification
- Using My Oracle Support for Additional Troubleshooting Information

1.1 Prerequisites to Integrating Oracle Identity Management Suite Components

Before using these procedures to integrate Identity Management components, you must install and deploy the components.

These prerequisites are explained in the following sections:

- Understanding the Installation Roadmap
- Understanding Deployment Topologies
- Understanding the Identity Store
- Understanding Integration Between LDAP Identity Store and Oracle Identity Governance
- Common Environment Variables
- Operating System

For details about installing Identity Management components, see About the Oracle Identity and Access Management Installation in *Installing and Configuring Oracle Identity and Access Management*.

1.1.1 Understanding the Installation Roadmap

You will take (or may already have taken) one of these paths in your IdM deployment:

- Installation, followed by component integration, and ending with scale-out (HA)
- Installation, followed by scale-out, and ending with integration

With scale-out, you may already have performed some of the integration procedures described here; notes in the relevant sections can help you determine whether a procedure is needed.

Introduction in the *Installing and Configuring Oracle Identity and Access Management* contains background on the IdM deployment procedure and describes the installation roadmap, prerequisites, and the installation and configuration workflow.

High Availability Concepts in the *High Availability Guide* explains the high availability solutions in Oracle Fusion Middleware, as well as the topologies and architecture of the various HA options.

1.1.2 Understanding Deployment Topologies

Before starting this integration, you must also understand the identity management topology and the environment in which the components will work together.

To learn more about the topology supported in this document, see Understanding Oracle Identity Management Integration Topologies.

1.1.3 Understanding the Identity Store

Oracle Identity Governance provides the ability to integrate an LDAP-based identity store into Oracle Identity Governance architecture. You can connect and manage an LDAP-based identity store directly from Oracle Identity Governance. Using this feature, you can use advanced user management capabilities of Oracle Identity Governance, including request-based creation and management of identities, to manage the identities within the corporate identity store.

In this deployment architecture, user identity information is stored in Oracle Identity Governance database to support the relational functionality necessary for Oracle Identity Governance to function, as well as in the LDAP store. All data is kept in sync transparently without the need for provisioning actions and setting up policies and rules. Identity operations started within Oracle Identity Governance, such as user creation or modification, are run on both the stores in a manner that maintains transactional integrity. In addition, any changes in the LDAP store made outside of Oracle Identity Governance are pulled into Oracle Identity Governance and made available as a part of the identity context.

1.1.4 Understanding Integration Between LDAP Identity Store and Oracle Identity Governance

Oracle Identity Governance users and roles are stored in the Oracle Identity Governance database. However, when a user, role, or role membership change takes place in Oracle Identity Governance, this information is propagated to the LDAP identity store. If a user, role, or role membership change takes place in LDAP directly, then these changes are synchronized into Oracle Identity Governance. The synchronization involves:

- Changes made in Oracle Identity Governance: User creation, modification, deletion, changes in enabled/disabled state and locked/unlocked states, and password changes are synchronized to LDAP.
- Role creation, modification, and deletion actions update the LDAP groups, including membership changes.

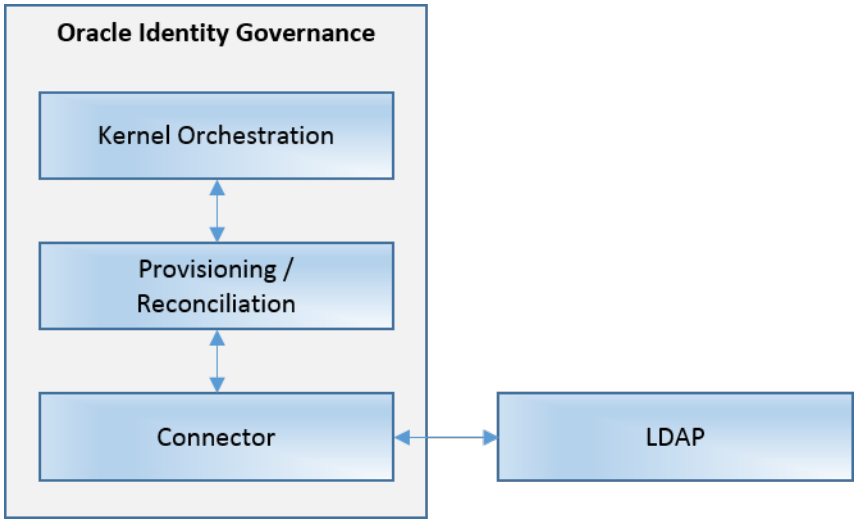
- Initial load of users, roles, and role memberships are synchronized.
- Direct changes to user profile in LDAP are reconciled to Oracle Identity Governance. However, a change to a user password made in LDAP is not reconciled to Oracle Identity Governance.
- Direct changes to roles and role memberships in LDAP are reconciled to Oracle Identity Governance.

When changes are made in the user and role data, the actual operation is performed with the help of the kernel handlers. These handlers go through an orchestration lifecycle of various stages, such as validation, preprocessing, action, and postprocessing.

Synchronization between Oracle Identity Governance and LDAP is performed by an LDAP connector library.

Figure 1-1 shows the communication between Oracle Identity Governance and LDAP.

Figure 1-1 Oracle Identity Governance and LDAP



1.1.5 Common Environment Variables

Shorthand notations are used to refer to common environment variables.

For example, the Oracle Middleware Home directory is often referred to as `MW_HOME`.

See "Identifying Installation Directories" in the *Installing and Configuring Oracle Identity and Access Management*.

1.1.6 Operating System

Currently, only Unix operating system is supported when integrating.

For details, see the note `Is Oracle Access Manager(OAM) Integrated With Oracle Identity Governance(OIG) Supported On The Windows Operating System(OS) (Doc ID 2780529.1)` at https://support.oracle.com .

1.2 Understanding Oracle Identity Management Integration Topologies

Oracle Identity Management consists of a number of products, which can be used either individually or collectively.

Two basic types of topology are available in Oracle Identity Management:

- Basic integration topology

  This topology supports integration between suite components, in an environment where each component runs on a separate node.

- Enterprise integration topology

  This topology supports integration between suite components in an enterprise environment. Each component may run on multiple nodes.

This book is dedicated to the first type, single-node integration topology. Use the procedures described in this book when deploying Oracle Identity Management in an environment where each component runs on its own node. You can also use the procedures to understand integration tools and techniques, and to understand the effects and benefits of integrating specific identity management components.

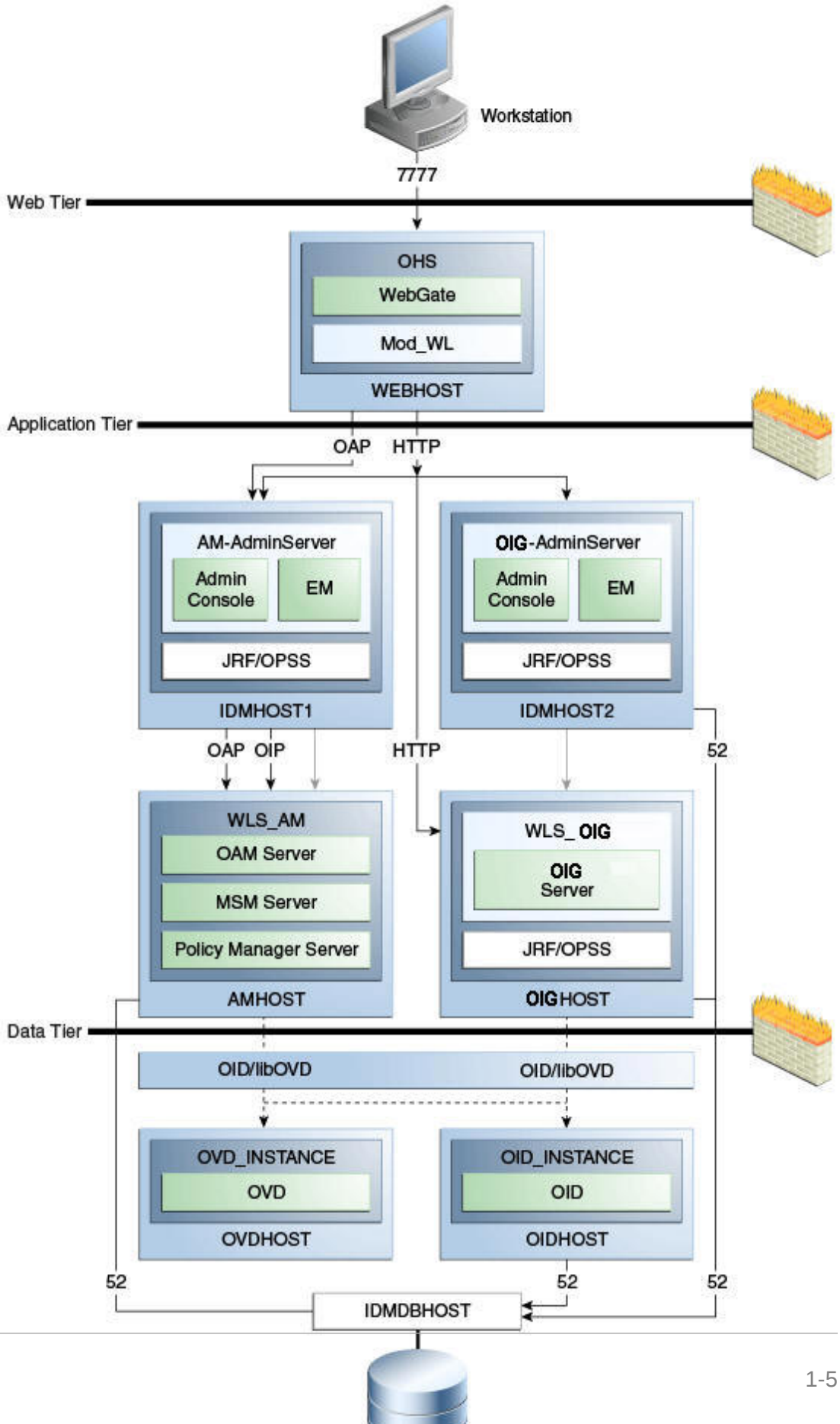
1.2.1 About the Basic Integration Topology

Basic integration topology is where the IdM components Access Manager and Oracle Identity Management are configured on separate Oracle WebLogic domains.

> **See Also:**
>
> Table 1-1 for definitions of acronyms used in this section.

Figure 1-2 shows a basic integration topology where the IdM components Access Manager and Oracle Identity Management are configured on separate Oracle WebLogic domains:

Figure 1-2 Basic Integration Topology with Multiple Administration Servers



Note that:

- All IdM components, including Access Manager server (`AMHOST`), the Oracle Identity Management server (`OIMHOST`), and Oracle Internet Directory (OID) are configured in separate WebLogic domains, and each is administered by its own administration server.

  Besides enhancing management of each component, this topology ensures you have flexibility when applying patches and upgrades. Patches for each component can be applied independently, with no version dependency on other components.
- For simplicity, some of the OMSS topology is omitted; for example the MSAS server which resides in the DMZ is not shown in the diagram.
- The BIP server and SOA Suite reside on the OIM domain; they are not shown in the diagram.
- The figure shows some representative ports only.

The SOA Suite used by OIM must be installed in the same domain as OIM. However, if you use SOA Suite for other purposes, you should consider setting up a separate install of SOA Suite for running your own services, composites, and other SOA features for that purpose.

In the single-domain architecture, Oracle Access Management Access Manager, Oracle Identity Management, and Oracle Mobile Security Access Server are configured on the same WebLogic domain. While possible, such a topology is not practical in the current context for the reasons cited above, and is not recommended for IdM integration.

> **See Also:**
>
> Overview of Oracle Identity Management Components Used in the Integration for an introduction to each IdM component.

1.2.1.1 About the Three Tier Architecture

This architecture can be viewed as consisting of three layers or zones:

- The Web Tier consists of the HTTP server and handles incoming Web traffic.
- The Application Tier contains identity management applications for managing identities and access, including Oracle Identity Management and Oracle Access Manager.
- The Data Tier, here considered to include the directory servers, hosts LDAPs and database.

1.2.1.2 Understanding the Web Tier

The web tier is in the DMZ Public Zone. The HTTP servers are deployed in the web tier.Most Identity Management components can function without the web tier. However, the web tier is required to support enterprise level single sign-on using products such as Access Manager.

The web tier is structured as follows in the single-node topology:

- `WEBHOST` has Oracle HTTP Server, WebGate (an Access Manager component), and the mod_wl_ohs plug-in module installed. The mod_wl_ohs plug-in module enables requests to be proxied from Oracle HTTP Server to a WebLogic Server running in the application tier.WebGate, an Access Manager component in Oracle HTTP Server, uses Oracle Access Protocol (OAP) to communicate with Access Manager running on `OAMHOST`. WebGate and Access Manager are used to perform operations such as user authentication.

1.2.1.3 Understanding the Application Tier

The application tier is the tier where Java EE applications are deployed. Products such as Oracle Identity Manager, Oracle Mobile Security Suite, Oracle Access Management Identity Federation, and Oracle Enterprise Manager Fusion Middleware Control are among key Java EE components deployed in this tier.

The Identity Management applications in the application tier interact with the directory tier as follows:

- They leverage the directory tier for enterprise identity information.
- They leverage the directory tier (and sometimes the database in the data tier) for application metadata.
- Fusion Middleware Control Console provides administrative functions to the components in the application and directory tiers.
- Oracle WebLogic Server has built-in web server support. If enabled, the HTTP listener exists in the application tier as well.

1.2.1.4 Understanding the Data Tier

The data tier is the deployment layer where all the LDAP services reside. This tier includes products such as Oracle Internet Directory (`OIDHOST`), Oracle Virtual Directory (`OVDHOST`), Oracle Unified Directory, and Oracle Database (`IDMDBHOST`).

The data tier stores two types of information:

- Identity Information: Information about users and groups resides in the identity store.
- Oracle Platform Security Services (OPSS): Information about security policies and about configuration resides in the policy store.

Policy information resides in a centralized policy store that is located within a database. You may store identity information in Oracle Internet Directory or in another directory.

If you store the identity details in a directory other than Oracle Internet Directory you can use Oracle Virtual Directory to present all the identity data in a single consolidated view that Oracle Identity Management components can interpret. See Configuring an Identity Store with Multiple Directories.

> **Note:**
>
> Oracle Identity Management uses Oracle Virtual Directory server or libOVD to access third-party directories.

1.2.2 About the Enterprise Integration Topology

Unlike single-node topologies, an enterprise integration topology takes into account such features as high availability, failover, and firewalls, and is beyond the scope of this document.

1.2.3 Using Multiple Directories for an Identity Store

Although the integration scenarios in this document focus on a simple identity store topology consisting of an Oracle Internet Directory LDAP server, your site may have some user data in a third-party directory, such as Microsoft Active Directory, and other user data in Oracle Internet Directory.

To account for this topology, you can use Oracle Virtual Directory to present all the identity data in a single consolidated view that Oracle Identity Management components can interpret.

See Configuring an Identity Store with Multiple Directories.

1.2.4 Integration Terminology

Definitions of terms that define the Oracle Fusion Middleware architecture.

Table 1-1 shows key terms and acronyms that are used to describe the architecture and topology of an Oracle Fusion Middleware environment:

Table 1-1 Oracle Fusion Middleware Integration Terminology

Term	Definition
IdM Configuration Tool	A command-line tool to verify the status of identity management components and to perform certain integration tasks.
Oracle Access Protocol (OAP)	A secure channel for communication between Webgates and Access Manager servers during authorization.
Oracle Fusion 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 HTTP Server (OHS)	Web server component for Oracle Fusion Middleware that provides a listener for Oracle WebLogic Server.
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 other Oracle home directories underneath the Middleware home directory.

Table 1-1 (Cont.) Oracle Fusion Middleware Integration Terminology

Term	Definition
Oracle home	An **Oracle home** contains installed files necessary to host a specific product. For example, the Oracle Identity Management Oracle home contains a directory that contains binary and library files for Oracle Identity Management. 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 instance	An **Oracle instance** contains one or more system components, such as Oracle Web Cache, Oracle HTTP Server, or Oracle Internet Directory. The system components in an Oracle instance must reside on the same machine. An Oracle instance directory contains files that can be updated, such as configuration files, log files, and temporary files. An Oracle instance is a peer of an Oracle WebLogic Server domain. Both contain specific configurations outside of their Oracle homes. The directory structure of an Oracle instance is separate from the directory structure of the Oracle home. It can reside anywhere; it need not be within the Middleware home directory.
Oracle WebLogic Server domain	A **WebLogic Server domain** is a logically related group of Java components. A WebLogic Server domain includes a special WebLogic Server instance called the Administration Server, which is the central point from which you configure and manage all resources in the domain. Usually, you configure a domain to include additional WebLogic Server instances called Managed Servers. You deploy Java components, such as Web applications, EJBs, and Web services, and other resources to the Managed Servers and use the Administration Server for configuration and management purposes only. Managed Servers in a WebLogic Server domain can be grouped together into a cluster. An Oracle WebLogic Server domain is a peer of an Oracle instance. Both contain specific configurations outside of their Oracle homes. The directory structure of an WebLogic Server domain is separate from the directory structure of the WebLogic Server home. It can reside anywhere; it need not be within the Middleware home directory.
system component	A **system component** is a manageable process that is not WebLogic Server. For example: Oracle HTTP Server, WebCache, and Oracle Internet Directory. Includes the JSE component.
Java component	A **Java component** is a peer of a system component, but is managed by the application server container. Generally refers to a collection of applications and resources, with generally a 1:1 relationship with a domain extension template. For example: SOA and WebCenter Spaces.
Oracle Fusion Middleware farm	Oracle Enterprise Manager Fusion Middleware Control is a Web browser-based, graphical user interface that you can use to monitor and administer an Oracle Fusion Middleware farm. An **Oracle Fusion Middleware farm** is a collection of components managed by Fusion Middleware Control. It can contain WebLogic Server domains, one or more Managed Servers and the Oracle Fusion Middleware system components that are installed, configured, and running in the domain.
Oracle Identity Management	The suite of identity and access management components in Oracle Fusion Middleware. See Overview of Oracle Identity Management Components Used in the Integration for details.

Table 1-1 (Cont.) Oracle Fusion Middleware Integration Terminology

Term	Definition
WebLogic Administration Server	The Administration Server is the central point from which you configure and manage all resources in the WebLogic domain.
WebLogic Managed Server	The Managed Server is an additional WebLogic Server instance to host business applications, application components, Web services, and their associated resources. Multiple managed servers can operate within the domain. Certain Managed Servers in the domain are created specifically to host Oracle Fusion Middleware components.

1.3 Overview of Oracle Identity Management Components Used in the Integration

This section provides a brief overview of Oracle Identity Management components whose integrations are described in this guide, and explains the benefits of integration.

Topics include:

- Oracle Unified Directory
- Oracle Internet Directory
- Oracle Access Management Access Manager
- Oracle Identity Governance
- Oracle Access Management Identity Federation

1.3.1 Oracle Unified Directory

Oracle Unified Directory is a comprehensive next generation directory service. It is designed to address large deployments and to provide high performance in a demanding environment.

The Oracle Unified Directory server is an LDAPv3-compliant directory server written entirely in Java. The directory server provides full LDAPv3 compliance, high performance and space effective data storage, and ease of configuration and administration.

Several procedures in this book feature Oracle Unified Directory as the repository for the identity store.

1.3.2 Oracle Internet Directory

Oracle Internet Directory is a general purpose directory service that enables fast retrieval and centralized management of information about dispersed users and network resources. It combines Lightweight Directory Access Protocol (LDAP) Version 3 with the high performance, scalability, robustness, and availability of an Oracle Database.

Oracle Internet Directory can serve as the repository for the identity store, which contains user identities leveraged by identity management components and other applications.

For details about integration with Oracle Internet Directory, see:

- Enabling LDAP Synchronization in Oracle Identity Governance

1.3.3 Oracle Virtual Directory

Oracle Virtual Directory, an LDAP version 3 enabled service that provides virtualized abstraction of one or more enterprise data sources into a single directory view. Oracle Virtual Directory makes many directories appear to be one local repository, hiding the complexity of data location, format, and protocol from client applications.

See Configuring Oracle Virtual Directory for Integration with Oracle Access Management Access Manager .

1.3.4 Oracle Access Management Access Manager

Oracle Access Management Access Manager provides a full range of Web perimeter security functions that include Web single sign-on; authentication and authorization; policy administration; auditing, and more. All existing access technologies in the Oracle Identity Management stack converge in Access Manager.

For details about integration with Access Manager, see:

- Integrating with Identity Federation

1.3.4.1 A Note About IDMDomain Agents and Webgates

By default, the IDMDomain Agent is enabled in the Oracle HTTP Server deployment. If you migrate from IDMDomain Agent to WebGate Agent, note the following:

- The protection policies set up for IDMDomain can be reused for WebGate if your webgate uses the IDMDomain preferredHost.
- IDMDomain and WebGate can coexist. If the IDMDomain Agent discovers a WebGate Agent in the Oracle HTTP Server deployment, IDMDomain Agent becomes dormant.

1.3.5 Oracle Identity Governance

Oracle Identity Management is a powerful and flexible enterprise identity management system that automatically manages users' access privileges within enterprise IT resources. Oracle Identity Manager is designed from the ground up to manage user access privileges across all of a firm's resources, throughout the entire identity management lifecycle—from initial creation of access privileges to dynamically adapting to changes in business requirements.

1.3.6 Oracle Access Management Identity Federation

To enhance support for federated authentication in cloud, web services, and B2B transactions, a SAML-based federation service is being introduced in a single access management server in 11*g* Release 2 (11.1.2). Oracle Access Management Identity Federation is an enterprise-level, carrier-grade service for secure identity information exchange between partners. Identity Federation protects existing IT investments by

integrating with a wide variety of data stores, user directories, authentication providers and applications.

In this initial release Identity Federation is limited to Service Provider mode. Identity Provider mode still requires an Oracle Identity Federation 11gR1 installation.

For details about using the Identity Federation service with Access Manager, see Integrating with Identity Federation.

1.4 IdM Integration Quick Links

Links to integration procedures.

Table 1-2 provides links to the integration procedures described here.

Table 1-2 Links to Integration Procedures in This Guide

Components to Integrate	Link
Post-install LDAP Synchronization with Oracle Identity Manager	Enabling LDAP Synchronization in Oracle Identity Governance
Oracle Virtual Directory and Oracle Identity Manager	Enabling LDAP Synchronization in Oracle Identity Governance
Oracle Virtual Directory and Access Manager	Configuring Oracle Virtual Directory for Integration with Oracle Access Management Access Manager
Access Manager and Oracle Identity Manager	Integrating Access Manager and Oracle Identity Governance
Access Manager and Identity Federation	Integrating with Identity Federation
Multi-Directory identity store	Configuring an Identity Store with Multiple Directories

Table 1-3 lists key integration procedures that appear in other IdM documents:

Table 1-3 Links to Integration Procedures in Other Guides

Components to Integrate	Link
Oracle Privileged Account Manager (OPAM) and Oracle Identity Manager (OIM)	Integrating with Oracle Identity Manager in *Oracle Fusion Middleware Administrator's Guide for Oracle Privileged Account Manager*
OPAM and OAM	Integrating with Oracle Access Management Access Manager in *Oracle Fusion Middleware Administrator's Guide for Oracle Privileged Account Manager*
OIM and Oracle Identity Analytics (OIA)	Integrating with Identity Analytics in *Administering Oracle Identity Governance*

1.5 Common Access Manager and Oracle Identity Governance Integration Scenarios

Common scenarios in Access Manager and Oracle Identity Management integrations for resource protection and password life cycle management are detailed here.

- About Password Management Scenarios
- About Managing Mobile Security Accounts and Applications Using Identity Self-Service

1.5.1 About Password Management Scenarios

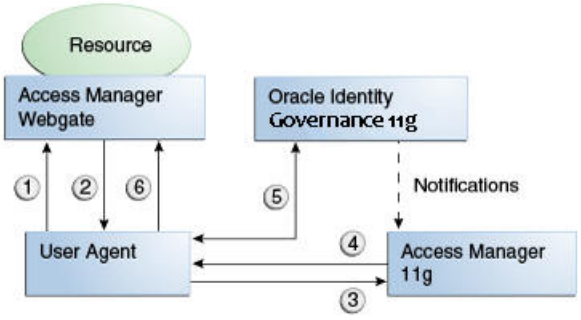
Common management scenarios supported by these deployment modes include:

- About Access Manager Integrated with Oracle Identity Governance
- About Self-Registration
- About Password Change
- About Forgot Password
- About Account Lock and Unlock
- About Challenge Setup

1.5.1.1 About Access Manager Integrated with Oracle Identity Governance

Figure 1-3 shows how password management is achieved when Access Manager and Oracle Identity Manager are integrated.

Figure 1-3 Integrating Access Manager and Oracle Identity Manager for Password Management



The flow of interactions between the components is as follows:

1. A user tries to access a resource protected by Access Manager.
2. The Oracle Access Management WebGate intercepts the (unauthenticated) request.
3. WebGate redirects the user to the Access Manager login service, which performs validation checks.

4. If Access Manager finds any password management trigger conditions, such as password expiry, it redirects users to Oracle Identity Manager.
5. Oracle Identity Manager interacts with the user to establish the user's identity and carry out the appropriate action, such as resetting the password.
6. Access Manager logs the user in by means of auto-login, and redirects the user to the Access Manager-protected resource which the user was trying to access in Step 1.

1.5.1.2 About Self-Registration

In this scenario, the user does not have an account but tries to access an Access Manager-protected resource. An Oracle Access Management 11*g* WebGate intercepts the request, detects that the user is not authenticated, and redirects the user to the Oracle Access Management Credential Collector (or 10*g* authenticating WebGate), which shows the Access Manager Login page containing a **Register New Account** link.

On selecting this link, the user is securely redirected to the Oracle Identity Manager Self Registration URL. Oracle Identity Manager interacts with the user to provision his account.

The Welcome Page is an unprotected page from which the self-registration/account creation can be initiated. This page contains two links, in addition to any introductory text or branding information. The links are:

- Register New Account - This is an unprotected URL to the corresponding application's registration wizard
- Login - This is a protected URL which serves as the landing page to which the user is directed after successfully completing the login.

> **Note:**
>
> Any application protected by a single sign-on system with the self-registration requirement is expected to support a self-registration page. The options are:
>
> - Self-registration using the default self-registration page or a customized version of the page.
>
> This is the most common option and is covered here.
> - Self-registration using anonymous pages in other applications.
>
> If the application dictates that the user be automatically logged in at the end of the registration process, it can implement this by using the Oracle Platform Security Services APIs.

The account creation flow is as follows:

1. The user (using his browser) accesses the application's welcome page, which contains a **Register New Account** link.
2. The user clicks the **Register New Account** link, which takes the user to a self-registration page provided by the application.
3. The user interacts with the application to self-register.

4. On completion, the application performs an auto-login for the user.

The protected application is expected to send an SPML request to Oracle Identity Manager to create the user. After this, the application could choose to do one of the following:

- The application may choose not to auto-login the user. The application redirects the user to the protected landing page URL. Access Manager then shows the login page and takes the user through the login flow.
- If there is no approval associated with the request, the application can make use of the Oracle Platform Security Services (OPSS) APIs to conduct an auto-login to the specific landing page URL and respond with a redirect request with that URL (along with the SSO cookie). This takes the user directly to the landing page without bringing up the login page.
- Auto-login cannot be done if approval is needed. The application determines which profile to use at the time of SPML request. The application needs to respond with an appropriate page indicating that the request has been submitted.

1.5.1.3 About Password Change

The Change Password flow enables users to change their password.

In the Change Password flow with Access Manager and Oracle Identity Manager, the user successfully logs into Access Manager but is required to immediately change the password. The user is not authorized to access protected resources until the password is changed and challenges have been set up.

On successful login, Access Manager detects if the triggering condition is in effect and redirects the user to the Oracle Identity Manager **Change Password** URL. Oracle Identity Manager facilitates the user password change or challenge set-up and resets the triggering condition.

On completion, Oracle Identity Manager redirects the user to the protected resource.

This situation is triggered in the following cases:

- The `Change Password upon Login` flag is on. This occurs:
  - when a new user is created
  - when the administrator resets a user's password
- The password has expired.

This flow describes the situation where a user logs in to an Access Manager-protected application for the first time, and is required to change password before proceeding.

The following describes the Change Password flow:

1. Using a browser, the user tries to access an application URL that is protected by Access Manager.
2. Oracle Access Management WebGate (SSO Agent) intercepts the request and redirects the user to the Access Manager Login Page.
3. The user submits credentials, which are validated by Access Manager.
4. Access Manager next determines if any of the First Login trigger conditions are valid. If so, Access Manager redirects the user to the Oracle Identity Manager Change Password URL.

5. Oracle Access Management WebGate (SSO Agent) intercepts the request, determines that Oracle Identity Manager is protected by the Anonymous Authentication Policy, and allows the user request to proceed.
6. Oracle Identity Manager interacts with the user to enable the user to change his password. On completion, Oracle Identity Manager updates the attributes that triggered the First Login flow. Oracle Identity Manager then performs a user auto-login.
7. Oracle Identity Manager notifies Access Manager of the successful first login.
8. Oracle Identity Manager redirects the user to the application URL the user tried to access in step 1.

1.5.1.4 About Forgot Password

The Forgot Password flow allows users to reset their password after successfully answering all challenge questions.

In this scenario, the user is at the Access Manager Login page and clicks the **Forgot Password** link. Access Manager redirects the user to the Oracle Identity Management **Forgot Password** URL, and passes the destination URL to which Oracle Identity Manager must redirect upon a successful password change as a query parameter (`backURL`).

Oracle Identity Management asks the user the challenge questions. Upon providing the correct responses, the user is allowed to specify a new password.

On completion, Oracle Identity Management redirects the user to the protected resource.

The Forgot Password flow is as follows:

1. Using a browser, the user tries to access an application URL that is protected by Access Manager.
2. The Oracle Access Management WebGate (SSO Agent) intercepts the request and redirects the user to the Access Manager Login Page.
3. The user clicks on the **Forgot Password** link on the Access Manager Login page, which sends the user to the Oracle Identity Manager **Forgot Password** URL.
4. Oracle Identity Manager interacts with the user to enable the user to reset the password. On completion, Oracle Identity Manager performs a user auto-login.
5. Oracle Identity Manager redirects the user to the application URL to which access was attempted in step 1.

1.5.1.5 About Account Lock and Unlock

Access Manager keeps track of login attempts and locks the account when the count exceeds the established limit in the password policy.

After the user account is locked, Access Manager displays the Help Desk contact information and Forgot Password link, or similar for any login attempt made. The information provided about the account unlocking process will need to be customized to reflect the process that is followed by your organization.

The following describes the account locking/unlocking flow:

1. Using a browser, a user tries to access an application URL that is protected by Access Manager.
2. Oracle Access Management WebGate (SSO Agent) intercepts the request and redirects the user to the Access Manager login page.
3. The user submits credentials that fail Access Manager validation. Access Manager renders the login page and asks the user to resubmit his or her credentials.
4. The user's unsuccessful login attempts exceed the limit specified by the policy. Access Manager locks the user account and redirects the user to the Access Manager Account Lockout URL. The resulting page displays the Help Desk contact information and Forgot Password link.
5. If the user contacts the Help Desk over the telephone and asks an administrator to unlock the account, then:
   a. The Help Desk unlocks the account using the Oracle Identity Manager administration console.
   b. Oracle Identity Manager notifies Access Manager of the account unlock event.
   c. The user attempts to access an application URL and this event triggers the normal Oracle Access Management single sign-on flow.
6. If the user uses the **Forgot Password** link, the user is sent to the Oracle Identity Manager Forgot Password URL, then:
   a. Oracle Identity Manager interacts with the user to enable the user to reset the password. On completion, Oracle Identity Manager performs a user auto-login.
   b. Oracle Identity Manager redirects the user to the application URL.

> **Note:**
>
> The user would be able to self-unlock the account by going through the Oracle Identity Manager Forgot Password flow, only once the user status is locked in Oracle Identity Manager. The user locked status is synchronized from the LDAP provider to Oracle Identity Manager only when the "LDAP User Create and Update Reconciliation" scheduled job is run.

1.5.1.6 About Challenge Setup

The Challenge Setup enables users to register challenge questions and answers.

When such redirection happens, Oracle Identity Management checks if the challenge questions are set. If not, it asks the user to set up challenge questions in addition to resetting the password.

Access Manager detects and redirects on password trigger conditions:

- Password Policy is updated to increase the required number of challenges.
- Password Policy is updated to require challenges

The following describes the flow:

> **Note:**
>
> The flow assumes First Login is not required.

1. Using a browser, the user tries to access an application URL that is protected by Access Manager.
2. Oracle Access Management WebGate (SSO agent) intercepts the request and redirects the user to the Access Manager Login Page.
3. The user submits credentials, which are validated by Access Manager. If a password triggering condition is detected, Access Manager redirects the user to the Oracle Identity Manager change password URL.
4. The Oracle Access Management WebGate (SSO agent) intercepts the request, determines that Oracle Identity Manager is protected by the anonymous authentication policy, and allows the user request to proceed.
5. Oracle Identity Manager interacts with the user to set up the challenges. On completion, Oracle Identity Manager updates the attributes that triggered the set challenges flow.
6. Oracle Identity Manager redirects the user to the application URL that the user attempted to access in Step 1.

1.5.2 About Managing Mobile Security Accounts and Applications Using Identity Self-Service

The Manage Mobile Security Account flow enables users to manage their mobile security accounts and applications. The flow between Oracle Mobile Security Suite and Oracle Mobile Security Suite-integrated components is as follows:

1. The user enrolls his mobile devices in Oracle Mobile Security Suite.
2. Oracle Mobile Security Suite provisions applications to the users based on his roles.
3. The user logs in to the Oracle Identity Governance Self Service Console to:
   - view his devices
   - perform operations, such as lock, wipe, or reset passcode for his device or workspace
4. The Oracle Mobile Security Suite task flows embedded in the Oracle Identity Management Console invokes Oracle Mobile Security Suite to obtain information on the devices and perform operations on them.

1.6 System Requirements and Certification

Refer to the system compatibility, requirements and certification documentation for information about hardware and software requirements, platforms, databases, and other information.

The compatibility documentation describes compatibility and interoperability considerations that may arise when you install, patch, or upgrade Oracle Fusion

Middleware 11g components. For details, see *Understanding Interoperability and Compatibility*.

The system requirements document covers information such as hardware and software requirements, minimum disk space and memory requirements, and required system libraries, packages, or patches.

The certification document covers supported installation types, platforms, operating systems, databases, JDKs, directory servers, and third-party products.

For the latest requirements and certification documentation refer to the table "Oracle Fusion Middleware Certification Matrices" in the *Understanding Interoperability and Compatibility*.

1.7 Using My Oracle Support for Additional Troubleshooting Information

You can use My Oracle Support (formerly MetaLink) to help resolve Oracle Fusion Middleware problems.

My Oracle Support contains several useful troubleshooting resources, such as:

- Knowledge base articles
- Community forums and discussions
- Patches and upgrades
- Certification information

> **Note:**
>
> You can also use My Oracle Support to log a service request.

You can access My Oracle Support at `https://support.oracle.com`.

Part II
Core Integrations

This part describes integrations between certain IdM components.

This part contains the following chapters:

2

Integrating Access Manager and Oracle Identity Governance

OAM-OIM integration using IDMConfigTool is not supported in 12*c*.

> **Important:**
>
> If you are upgrading from 11.1.2.3 to 12.2.1.3, you can continue with the OAM-OIM integration procedure mentioned in the previous release of the suite-level Integration Guide. See Integrating Access Manager and Oracle Identity Manager.

3

Integrating Access Manager, OAAM, and Oracle Identity Governance

OAM-OAAM-OIM integration is not supported in 12*c*. However, the setup with OAM and OIM upgraded from 11.1.2.3 to 12.2.1.3 along with OAAM in 11.1.2.3 is supported.

> **Important:**
>
> If you have OAM-OIM-OAAM integrated in 11*g* environment, and you are upgrading OIM and OAM from 11.1.2.3 to 12.2.1.3, then you can continue with the OAM-OAAM-OIM integration procedure mentioned in the previous release of the suite-level Integration Guide . See Integrating Access Manager, OAAM, and OIM.

Part III
External SSO Solutions

You can integrate federation partners into the Oracle IdM environment.

This part contains the following chapter:

- Integrating with Identity Federation

4

Integrating with Identity Federation

This chapter explains how Oracle Access Management Access Manager leverages identity federation to create an authenticated session with a federation partner.
This chapter contains these sections:

- Introduction to Identity Federation with Oracle Access Manager
- Integrating Access Manager 11gR2 with Identity Federation 11gR1
- Running Access Manager-OIF Integration Scripts to Automate Tasks

4.1 Introduction to Identity Federation with Oracle Access Manager

This section provides background about federation with Access Manager.

Topics include:

- About Oracle Access Management Identity Federation
- About Deployment Options for Identity Federation

4.1.1 About Oracle Access Management Identity Federation

Identity federation is available in two architectures:

- As a federation engine, known as Oracle Access Management Identity Federation, built into Oracle Access Management (11g Release 2 (11.1.2).
- As a standalone, self-contained federation server, known as Oracle Identity Federation, that enables single sign-on and authentication in a multiple-domain identity network (11g Release 1 (11.1.1).

  The SP integration Engine included with Oracle Identity Federation consists of a servlet that processes requests from the server to create a user authenticated session at the Identity and Access Management (IAM) server. The engine includes several internal plug-ins that allow it to interact with different IAM servers, including Access Manager (formerly Oracle Access Manager).

4.1.2 About Deployment Options for Identity Federation

> **See Also:**
>
> For details about naming conventions and name changes in Oracle Access Management, see *Introduction to Oracle Access Management* in *Administering Oracle Access Management*.

Various deployment options are available for leveraging identity federation with Access Manager to create an authenticated user session.

The Oracle Fusion Middleware framework supports these integrated approaches to cross-domain single sign-on:

- An Oracle Access Management Identity Federation engine built into the Access Manager server. All configuration is performed in Access Manager.

  This approach is available in 12*c* (12.2.2). The engine supports both Service Provider (SP) and Identity Provider (IdP) modes.

- Separate Oracle Identity Federation and Oracle Access Manager servers that can be integrated to provide federation capabilities. Management and configuration of both servers is required for this integration.

  This approach is available in 11*g* Release 1 (11.1.1).

  Under this approach, Oracle Identity Federation provides two deployment scenarios for Oracle Access Manager:

  – Oracle Identity Federation 11*g* Release 1 (11.1.1) integrated with Oracle Access Manager 10*g*

  – Oracle Identity Federation 11*g* Release 1 (11.1.1) integrated with Access Manager 11*g*

Table 4-1 summarizes the options available to integrate the identity federation products with Oracle Access Management Access Manager and provides links to deployment procedures:

Table 4-1 Deployment Options involving Oracle Access Manager 10g and Access Manager 11g

Access Manager Version	Description	Additional Information
Oracle Access Management Access Manager 11*g*R2	Access Manager contains a built-in federation engine that supports both SP and IdP mode functionality configurable through the Oracle Access Management Console.	Introduction to Federation within Oracle Access Suite Console in *Administering Oracle Access Management* Integrating Access Manager 11gR2 with Identity Federation 11gR1
Oracle Access Manager 11*g*R1	The stand-alone Oracle Identity Federation 11g Release 1 server integrates with the Access Manager 11*g* server.	Integrating Oracle Identity Federation in Integration Guide for Oracle Access Manager
Oracle Access Manager 10*g*	The stand-alone Oracle Identity Federation 11g Release 1 server integrates with the Oracle Access Manager 10*g* server.	Deploying Oracle Identity Federation with Oracle Access Manager 10g in Oracle® Fusion Middleware Administrator's Guide for Oracle Identity Federation

4.2 Integrating Access Manager 11gR2 with Identity Federation 11gR1

This section describes how to integrate Access Manager 12*c* (12.2.2) with Oracle Identity Federation 11*g* Release 1 (11.1.1).

This is also referred to as Access Manager 11*g*R2 with Oracle Identity Federation 11*g*R1.

4.2.1 About SP and Authentication Integration Modes

Two integration modes are described in this chapter:

- SP Mode

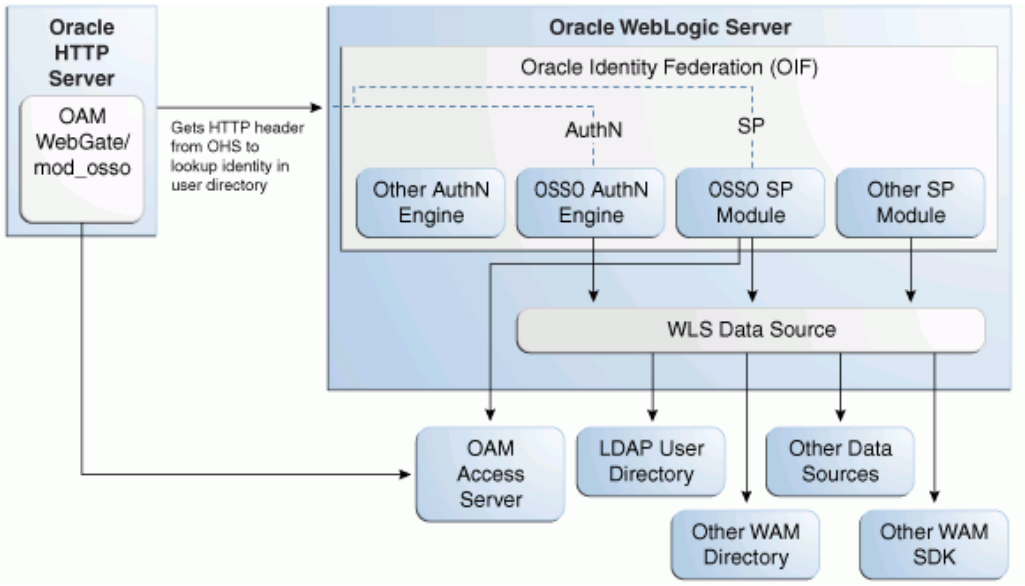
  This mode enables Oracle Identity Federation to authenticate the user via Federation SSO and propagate the authentication state to Access Manager, which maintains the session information.

- Authentication Mode

  This mode enables Access Manager to authenticate the user on behalf of Oracle Identity Federation.

Figure 4-1 describes the processing flow in each mode:

Figure 4-1 Access Manager with Identity Federation



In the SP mode, Oracle Identity Federation uses the federation protocols to identify a user, and requests Access Manager to create an authenticated session at Access Manager.

In the authentication mode, Oracle Identity Federation delegates authentication to Access Manager through the use of a WebGate agent protecting an Oracle Identity Federation resource. Once the user is authenticated, the WebGate will assert the user's identity by an HTTP Header that Oracle Identity Federation will read to identify the user.

4.2.2 Access Manager and Oracle Identity Federation Integration Overview

The integration between Access Manager and Oracle Identity Federation requires the following tasks:

- Ensure that the necessary components, including Oracle WebLogic Server and Identity Management (IdM) components, are installed and operational. For details, see Verifying Servers are Running and a Resource is Protected.
- Register Oracle HTTP Server as a partner with Access Manager to protect a resource. For details, see Registering Oracle HTTP Server WebGate with Access Manager for Access Manager and OIF Integration.
- Configure the Oracle Identity Federation server to function as a service provider (SP) and/or as an identity provider (IdP) with Access Manager. For details, see Configuring Oracle Identity Federation for Access Manager and OIF Integration.
- Configure Access Manager to delegate authentication to Oracle Identity Federation and/or to authenticate a user on behalf of Oracle Identity Federation, For details, see Configuring Access Manager for Integration with Oracle Identity Federation.

4.2.3 Prerequisites to Integrating Access Manager with Oracle Identity Federation

You must install the following components prior to undertaking the integration tasks:

- Oracle WebLogic Server
- Oracle HTTP Server 11*g*
- Access Manager 11*g*
- Oracle Identity Federation 11*g*
- WebGate (required in authentication mode)

> **Note:**
>
> Refer to the Certification Matrix for platform and version details.

> **See Also:**
>
> *Oracle® Fusion Middleware Installation Guide for Oracle Identity Manager*

4.2.4 Verifying Servers are Running and a Resource is Protected

Check the following components before starting the configuration process:

- Oracle WebLogic Server

  Ensure that the administration and managed servers are up and running.

- Oracle HTTP Server

  For testing purposes, identify or create a resource to be protected. For example, create an `index.html` file to serve as a test resource.

- Oracle Identity Federation

  Access the Fusion Middleware Control console for the Oracle Identity Federation server using a URL of the form:

  `http://oif_host:oif_em_port/em`

Verify that all the servers are running.

4.2.5 Registering Oracle HTTP Server WebGate with Access Manager for Access Manager and OIF Integration

This section shows how you can register Oracle HTTP Server and 11g WebGate with Access Manager, depending on the protection mechanism you have chosen.

Follow these steps to register Oracle HTTP Server and Access Manager 11g WebGate with Access Manager for authentication:

> **Note:**
>
> In this procedure, `MW_HOME` represents the Oracle Fusion Middleware Home directory.

1. Locate the `OAM11GRequest.xml` file or the `OAM11GRequest_short.xml` file, which resides in the directory:

   ```
   MW_HOME/Oracle_IDM1/oam/server/rreg/input
   ```

2. Make the necessary changes to the file.
3. Locate the `oamreg.sh` script, which resides in the directory:

   ```
   MW_HOME/Oracle_IDM1/oam/server/rreg/bin
   ```

4. Execute the script using the command string:

   > **Note:**
   >
   > The user is `weblogic`, and you must supply the password.

   ```
   ./oamreg.sh inband input/OAM11GRequest.xml
   ```

   or

   ```
   ./oamreg.sh inband input/OAM11GRequest_short.xml
   ```

5. Using the Oracle Access Management Console, create a resource representing the Oracle Identity Federation URL to be protected by Access Manager for authentication. This URL contains the hostname and port of the Oracle Identity Federation server, and the path to the resource, which is mode-dependent:

   ```
   http(s)://oif-host:oif-port/fed/user/authnoam11g
   ```

6. Protect this resource with an authentication policy and an authorization policy.
7. Restart Oracle HTTP Server:

   ```
   Oracle_WT1/instances/instance1/bin/opmnctl restartproc process-type=OHS
   ```

   You can also restart Oracle HTTP Server with:

   ```
   Oracle_WT1/instances/instance1/bin/opmnctl stopall
   Oracle_WT1/instances/instance1/bin/opmnctl startall
   ```

4.2.6 Configuring Oracle Identity Federation for Access Manager and OIF Integration

This section describes how to configure Oracle Identity Federation to be integrated with Access Manager:

- In SP mode, Access Manager will delegate authentication to Oracle Identity Federation for Federation SSO.
- In Authentication mode, Oracle Identity Federation will delegate authentication to Access Manager.

This section contains these topics:

- Verifying the Oracle Identity Federation User Data Store
- Configuring the Oracle Identity Federation Authentication Engine
- Configuring the Oracle Identity Federation SP Integration Module

4.2.6.1 Verifying the Oracle Identity Federation User Data Store

Oracle Identity Federation and Access Manager must use the same LDAP directory:

- The LDAP directory to be used must be defined in Access Manager as the default Identity Store.
- The Oracle Identity Federation User Data Store must reference the LDAP directory to be used.

Take these steps to verify the data store configuration:

1. Locate the Oracle Identity Federation instance in Fusion Middleware Control.
2. Navigate to **Administration**, then **Data Stores**.
3. Ensure that the user data store points to the same directory as the default Access Manager identity store.

4.2.6.2 Configuring the Oracle Identity Federation Authentication Engine

> **Note:**
>
> Running Access Manager-OIF Integration Scripts to Automate Tasks describes scripts that you can execute to automatically perform the manual operations shown here.

Take these steps to configure the Oracle Identity Federation Authentication Engine to retrieve information provided by the WebGate 11g agent:

1. Locate the Oracle Identity Federation instance in Fusion Middleware Control.
2. Navigate to **Administration**, then **Authentication Engines**.
3. Enable the Access Manager 11g authentication engine.
4. Select WebGate 11g as the Agent Type.
5. Enter `OAM_REMOTE_USER` as the User Unique ID Header.
6. In the Default Authentication Engine drop-down list, select Oracle Access Manager 11g.
7. Configure logout:

- If Oracle Identity Federation is also going to be integrated with Access Manager in SP mode, then disable logout as the logout integration with Access Manager 11g will be performed with the OAM11g SP engine.
- If Oracle Identity Federation is not going to be integrated with Access Manager in SP mode:
  - Enable logout
  - Enter the following as the URL:

    ```
    http(s)://oam_host:oam_port/oam/server/logout
    ```

8. Click **Apply**.

4.2.6.3 Configuring the Oracle Identity Federation SP Integration Module

This section lists the steps that need to be performed to configure Oracle Identity Federation in SP mode for Access Manager, so that Oracle Identity Federation can send assertion tokens and direct session management to Access Manager.

> **Note:**
>
> Running Access Manager-OIF Integration Scripts to Automate Tasks describes scripts that you can execute to automatically perform the manual operations shown here.

The steps to achieve this are as follows:

1. Locate the Oracle Identity Federation instance in Fusion Middleware Control.
2. Navigate to Administration, then Service Provider Integration Modules.
3. Select the Oracle Access Manager 11g tab.
4. Configure the page as follows:
   - Check the **Enable SP Module** box.
   - In the Default SP Integration Module drop-down, select Oracle Access Manager 11g.
   - Check the **Logout Enabled** box.
   - Configure these URLs:

     ```
     Login URL :  http(s)://oam_host:oam_port/oam/server/dap/cred_submit
     Logout URL:  http(s)://oam_host:oam_port/oam/server/logout
     ```

     where *oam_host* and *oam_port* are the host and port number of the Access Manager server respectively.
   - Set Username Attribute value to "cn" to match the Access Manager username attribute.
   - Click **Apply**.
5. Click **Regenerate**.

   This action generates a keystore file that contains the keys used to encrypt and decrypt the tokens that are exchanged between the Access Manager and Oracle

Identity Federation servers. Be sure to save the keystore file using the **Save As** dialog.

Copy the keystore file to a location within the installation directory of Access Manager.

> **Note:**
>
> Make a note of the location, since you will need to refer to it later.

4.2.7 Configuring Access Manager for Integration with Oracle Identity Federation

This section describes how to configure Access Manager to integrate with Oracle Identity Federation:

- In SP mode, Access Manager will delegate authentication to Oracle Identity Federation for Federation SSO.
- In Authentication mode, Oracle Identity Federation will delegate authentication to Access Manager.

This section contains these topics:

- Configuring Access Manager to Redirect Users to Oracle Identity Federation
- Registering Oracle Identity Federation as a Trusted Access Manager Partner

4.2.7.1 Configuring Access Manager to Redirect Users to Oracle Identity Federation

This task configures Access Manager to redirect the user to Oracle Identity Federation for authentication when `OIFScheme` is used to protect a resource using Federation single sign-on. The steps needed to achieve this are as follows:

1. Log in to the Oracle Access Management Console:

   `http://`*`oam_adminserver_host:oam_adminserver_port`*`/oamconsole`
2. Select the Policy Configuration tab.
3. Select and open the `OIFScheme`.
4. In the Challenge URL field, modify the value of `OIF-Host` and `OIF-Port`:

   `http(s)://`*`oif-host:oif-port`*`/fed/user/spoam11`
5. Confirm that the value of the Context Type drop-down is set to "external".
6. Click **Apply** to save the changes.

4.2.7.2 Registering Oracle Identity Federation as a Trusted Access Manager Partner

If Oracle Identity Federation is used in SP mode only, or authentication and SP mode, refer to Registering Oracle Identity Federation for Use in SP Mode.

If Oracle Identity Federation is used in authentication mode only, refer to Registering Oracle Identity Federation for Use in Authentication Mode.

> **Note:**
>
> Running Access Manager-OIF Integration Scripts to Automate Tasks describes scripts that you can execute to automatically perform the manual operations shown here to register Oracle Identity Federation as a trusted partner.

4.2.7.2.1 Registering Oracle Identity Federation for Use in SP Mode

> **Note:**
>
> Prior to performing this procedure, ensure that OAM Admin Server and all Managed Servers are running.

Copy the keystore file to a directory under the middleware home in which the Access Manager server is installed.

Use a WLST command to update the `OIFDAP` partner block in the `oam-config.xml` configuration file. The steps and syntax are as follows:

1. Enter the shell environment by executing:

   ```
   $DOMAIN_HOME/common/bin/wlst.sh
   ```

2. Connect to the Access Manager administration server with the following command syntax:

   ```
   connect('weblogic','password','host:port')
   ```

3. Execute the command to update the partner block in the configuration file:

   ```
   registerOIFDAPPartner(keystoreLocation=location of keystore file,
   logoutURL=logoutURL)
   ```

   where `logoutURL` is the Oracle Identity Federation logout URL that is invoked when the Access Manager server logs out the user.

   For example:

   ```
   registerOIFDAPPartner(keystoreLocation="/home/pjones/keystore",
   logoutURL="http://abcdef0123.in.mycorp.com:1200/fed/user/spslooam11g?
   doneURL=http://abc1234567.in.mycorp.com:6001/oam/pages/logout.jsp")
   ```

4.2.7.2.2 Registering Oracle Identity Federation for Use in Authentication Mode

Use a WLST command to update the `OIFDAP` partner block in the `oam-config.xml` configuration file. The steps and syntax are as follows:

1. Enter the shell environment by executing:

   ```
   $DOMAIN_HOME/common/bin/wlst.sh
   ```

2. Connect to the Access Manager administration server with the following command syntax:

```
connect('weblogic','password','host:port')
```

3. Execute the command to update the partner block in the configuration file:

```
registerOIFDAPPartnerIDPMode(logoutURL=logoutURL)
```

where `logoutURL` is the Oracle Identity Federation logout URL that is invoked when the Access Manager server logs out the user.

For example:

```
registerOIFDAPPartnerIDPMode(logoutURL="http://abcdef0123.in.mycorp.com:1200/fed/
user/authnslooam11g?doneURL=http://abc1234567.in.mycorp.com:6001/oam/pages/
logout.jsp")
```

4.2.8 Configuring Access Manager to Protect a Resource with the OIFScheme

After the integration of Access Manager and Oracle Identity Federation in SP mode, a resource can now be protected with `OIFScheme`, which will trigger a Federation single sign-on operation when an unauthenticated user requests access to a resource protected by that scheme.

In an Application Domain of the Policy Configuration tab, define an Authentication Policy using the `OIFScheme`, and protect a resource with that authentication policy.

4.2.9 Testing the Access Manager and Oracle Identity Federation Integration Configuration

The final configuration task is to test whether the integration is correctly configured. The steps differ between authentication mode and SP mode.

- Testing the SP Mode Configuration
- Testing the Authentication Mode Configuration

4.2.9.1 Testing the SP Mode Configuration

Take these steps to test for correct configuration in SP mode:

1. Establish federated trust between Oracle Identity Federation and a remote Identity Provider (IdP).
2. Set that identity provider as the default SSO identity provider.
3. Try accessing the protected resource.
4. When set up correctly, you should be redirected to the IdP for authentication. Verify that user credentials are required on this page.
5. Enter valid credentials on the login page.

> **Note:**
>
> The user should exist in both the IdP security domain and the Oracle Identity Federation/Access Manager security domain.

6. Check that you are redirected to the protected page.
7. Verify that the following cookies are created:
   - `OAM_ID`
   - `ORA_OSFS_SESSION`
   - `OHS Cookie`

4.2.9.2 Testing the Authentication Mode Configuration

Take these steps to test for correct configuration in authentication mode:

1. Establish federated trust between Oracle Identity Federation and a remote service provider.
2. Initiate federation single sign-on from the service provider.
3. Verify that you are redirected to the Access Manager login page at the IdP. On this page user credentials are requested.
4. Enter the relevant credentials and process the page.
5. Verify that you are redirected to the service provider domain.

4.3 Running Access Manager-OIF Integration Scripts to Automate Tasks

The automated steps make the integration smoother and faster than a purely manual procedure.

This section describes scripts that automate some of the Oracle Identity Federation configuration tasks described in Integrating Access Manager 11gR2 with Identity Federation 11gR1 for Oracle Access Manager integration.

This section contains these topics:

- Performing Prerequisite Steps Before Integration
- Verifying WebLogic and Oracle Identity Federation Servers are Running
- Executing the Automated Procedure for Access Manager-OIF Integration

4.3.1 Performing Prerequisite Steps Before Integration

The prerequisite procedure is performed before you do anything else for integration. Ensure that the following have been done:

1. The following components are installed:
   - Oracle WebLogic Server
   - Oracle HTTP Server
   - Oracle Access Manager 11*g*
   - Oracle Identity Federation 11*g*

> **Note:**
>
> Refer to the Certification Matrix for platform and version details.

For guidance on integration prerequisites, see *Installing and Configuring Oracle Internet Directory*.

2. Oracle Identity Federation 11g and OHS are integrated; that is, OHS is configured as the front end to the Oracle Identity Federation server.

   For details, see "Deploying Oracle Identity Federation with Oracle HTTP Server" in the *Oracle® Fusion Middleware Administrator's Guide for Oracle Identity Federation*.

3. The SSO agent is already created and integrated with Access Manager 11g.

4.3.2 Verifying WebLogic and Oracle Identity Federation Servers are Running

Verify WebLogic and Oracle Identity Federation Servers are running.

- Oracle WebLogic Server

  Ensure that the administration and managed servers are up and running.

- Oracle Identity Federation

  Access the Fusion Middleware Control console for the Oracle Identity Federation server using a URL of the form:

  `http://oif_host:oif_em_port/em`

Verify that all the servers are running.

4.3.3 Executing the Automated Procedure for Access Manager-OIF Integration

Automating some tasks in the integration of Access Manager with Oracle Identity Federation is achieved by executing python scripts provided in the distribution.

Configuring Oracle Identity Federation for Access Manager and OIF Integration describes the tasks that you can automate with scripts.

- Tasks Performed by Federation Configuration Scripts
- Copying the Access Manager-OIF Integration Scripts to the Access Manager Machine
- Understanding Inputs to the Access Manager-OIF Integration Scripts
- Running the Access Manager-OIF Integration Scripts

4.3.3.1 Tasks Performed by Federation Configuration Scripts

The scripts perform the following tasks/procedures:

- Automation of all Oracle Identity Federation configuration
- Registration of Oracle Identity Federationas DAP partner in Access Manager

- Addition of Oracle Identity Federation URLs as protected resources in the policy domain.

4.3.3.2 Copying the Access Manager-OIF Integration Scripts to the Access Manager Machine

You need to copy certain files to the Access Manager host. The files are as follows:

- `setupOIFOAMConfig.sh`,
- `setupOIFOAMIntegration.py`
- locale specific resource bundle `oifWLSTResourceBundle_`*locale*`.properties`

Create a directory to save these files or copy into an existing directory, in the Access Manager host machine. For example, `/scratch/scripts` (linux) or `c:\temp\scripts` (Windows).

4.3.3.3 Understanding Inputs to the Access Manager-OIF Integration Scripts

The script takes in named parameters as inputs (order of inputs does not matter). The inputs mostly have default values if not passed in.

Table 4-2 shows the inputs needed by the scripts:

Table 4-2 Inputs for the Access Manager-OIF 11gR1 Integration Scripts

Parameter	Description	Default	Required?
oifHost	Hostname of Oracle Identity Federation managed server	None	Yes
oifPort	Port number of Oracle Identity Federation Managed server	7499	No
oifAdminHost	Hostname of Oracle Identity Federation Admin server	oifHost	No
oifAdminPort	Port number of Oracle Identity Federation Admin server	7001	No
oamAdminHost	Hostname of Access Manager Admin server	localhost	No
oamAdminPort	Port number of Access Manager Admin server	7001	No
agentType	Agent type used, such as webgate10g, webgate11g, mod_osso	webgate11g	No

> **Note:**
>
> The agent type is the agent created in Access Manager using the `rreg` tool or through the Oracle Access Management Console.

4.3.3.4 Running the Access Manager-OIF Integration Scripts

The automation is run by executing the script file `setupOIFOAMConfig.sh` (Linux) or `setupOIFOAMConfig.cmd` (Windows).

The steps are as follows:

On Unix:

The following steps show how to run the script. Substitute the sample parameter values with appropriate values.

1. In a command line prompt set the `DOMAIN_HOME`:

   ```
   export DOMAIN_HOME=path to domain home
   ```

2. If Oracle Identity Federation administration and managed server are on the same host and the agent type is non-default (for example, webgate10g), execute the command:

   ```
   ./setupOIFOAMConfig.sh oifHost=myhost oifPort=portnum oamAdminHost=myhost2
   oamAdminPort=portnum2 agentType=webgate10g
   ```

3. If Oracle Identity Federation administration and managed server are on different hosts, with a default agent type (webgate11g), execute the command:

   ```
   ./setupOIFOAMConfig.sh oifHost=myhost oifPort=portnum oifAdminHost=myhost2
   oifAdminPort=portnum2 oamAdminHost=myhost3 oamAdminPort=portnum3
   ```

4. If Oracle Identity Federation administration and managed server are on the same host, and all defaults apply from Table 4-2, execute the command:

   ```
   ./setupOIFOAMConfig.sh oifHost=myhost oamAdminHost=myhost2
   ```

On Windows:

The following steps show how to run the script. Substitute the sample parameter values with appropriate values.

1. In a command line prompt set the `DOMAIN_HOME`:

   ```
   set DOMAIN_HOME=path to oam domain home
   ```

2. If Oracle Identity Federation administration and managed server are on the same host and the agent type is non-default (for example, webgate10g), execute the command:

   ```
   setupOIFOAMConfig.cmd  "oifHost=myhost" "oifPort=portnum" "oamAdminHost=myhost2"
   "oamAdminPort=portnum2" "agentType=webgate10g"
   ```

3. If Oracle Identity Federation administration and managed server are on different hosts, with a default agent type (webgate11g), execute the command:

   ```
   setupOIFOAMConfig.cmd "oifHost=myhost" "oifPort=portnum" "oifAdminHost=myhost2"
   "oifAdminPort=portnum2" "oamAdminHost=myhost3" "oamAdminPort=portnum3"
   ```

4. If Oracle Identity Federation administration and managed server are on the same host, and all defaults apply from Table 4-2, execute the command:

   ```
   setupOIFOAMConfig.cmd "oifHost=myhost" " "oamAdminHost=myhost3"
   ```

Part IV
Additional Identity Store Configuration

This part contains topics related to additional configuration of the identity store.

This part contains the following chapter:

- Configuring an Identity Store with Multiple Directories

5

Configuring an Identity Store with Multiple Directories

This chapter explains how to prepare directories other than Oracle Internet Directory for use as an Identity Store.
This chapter contains the following topics:

5.1 Overview of Configuring Multiple Directories as an Identity Store

This chapter describes how to configure Oracle Virtual Directory for two multiple directory scenarios. In both scenarios, you have some user data in a third-party directory, such as Active Directory, and other user data in Oracle Internet Directory.

In both scenarios, you use Oracle Virtual Directory to present all the identity data in a single consolidated view that Oracle Identity Management components can interpret.

The scenarios are as follows:

- **Split Profile**: A split profile, or split directory configuration, is one where identity data is stored in multiple directories, possibly in different locations. You use a split profile when you must extend directory schema in order to support specific schema elements, but you cannot or do not want to extend the schema in the third-party Identity Store. In that case, deploy an Oracle Internet Directory as a shadow directory to store the extended attributes. For details, see Configuring Multiple Directories as an Identity Store: Distinct User and Group Populations in Multiple Directories . (If, on the other hand, you can extend the schema, use the approach described in Section 2.2.3, "Extending the Directory Schema for Access Manager.")
- **Distinct User and Group Populations**: Another multidirectory scenario is one where you have distinct user and group populations, such as internal and external users. In this configuration, Oracle-specific entries and attributes are stored in Oracle Internet Directory. Enterprise-specific entries, for example, entries with Fusion Applications-specific attributes, are stored in Active Directory. For details, see Configuring Multiple Directories as an Identity Store: Distinct User and Group Populations in Multiple Directories .

In this chapter, Active Directory is chosen as the non-Oracle Internet Directory Enterprise Directory. The solution is applicable to all enterprises having one or more Active Directories as their enterprise Identity Store.

5.2 Configuring Multiple Directories as an Identity Store: Split Profile

This section describes how to configure multiple directories as an Identity Store. In cases where the Active Directory schema cannot be extended, you use Oracle Internet Directory as a shadow directory to store these attributes. Oracle Virtual Directory links them together to present a single consolidated DIT view to clients. This is called a split profile or split directory configuration. In this configuration, all the Oracle specific attributes and Oracle specific entities are created in Oracle Internet Directory.

This section contains the following topics:

5.2.1 Prerequisites to Configuring Multiple Directories as an Identity Store

The following assumptions and rules apply to this deployment topology:

- Oracle Internet Directory houses the Fusion Identity Store. This means that Oracle Internet Directory is the store for all Fusion Application-specific artifacts. The artifacts include a set of enterprise roles used by Fusion Application and some user attributes required by Fusion Applications. All other stores are referred to as enterprise Identity Stores.
- The enterprise contains more than one LDAP directory. Each directory contains a distinct set of users and roles.
- The enterprise policy specifies that specific user attributes, such as Fusion Application-specific attributes, cannot be stored in the enterprise directory. All the extended attributes must be stored in a separate directory called the shadow directory. This shadow directory must be Oracle Internet Directory because Active Directory does not allow you to extend the schema.
- User login IDs are unique across the directories. There is no overlap of the user login IDs between these directories.
- Oracle Identity Management has no fine-grained authorization. If Oracle Identity Management's mapping rules allow it to use one specific subtree of a directory, then it can perform all CRUD (Create, Read, Update, Delete) operations in that subtree of the LDAP directory. There is no way to enable Oracle Identity Management to read user data in a subtree but not enable it to create a user or delete a user in subtree.

- Referential integrity must be turned off in Oracle Internet Directory so that an Oracle Internet Directory group can have members that are in one of the Active Directory directories. The users group memberships are not maintained across the directories with referential integrity.

5.2.2 Repository Descriptions

This section describes the artifacts in the Identity store and how they can be distributed between Active Directory and Oracle Internet Directory, based on different enterprise deployment requirements.

The Artifacts that are stored in the Identity Store are:

- Application IDs: These are the identities that are required to authenticate applications to communicate with each other.
- Seeded Enterprise Roles: These are the enterprise roles or LDAP group entries that are required for default functionality.
- Enterprise roles provisioned by Oracle Identity Management: These are runtime roles.
- Enterprise Users: These are the actual users in the enterprise.
- Enterprise Groups: These are the roles and groups that already exist in the enterprise.

In a split profile deployment, the Identity Store artifacts can be distributed among Active Directory and Oracle Internet Directory, as follows.

- Oracle Internet Directory is a repository for enterprise roles. Specifically, Oracle Internet Directory contains the following:
  - Application IDs
  - Seeded enterprise roles
  - Enterprise roles provisioned by Oracle Identity Management
- Active Directory is the repository for:
  - Enterprise users
  - Enterprise groups (not visible to Oracle Identity Management or Fusion Applications)

The following limitations apply:

- The Active Directory users must be members of Oracle Internet Directory groups.
- The groups in Active Directory are not exposed at all. Oracle applications only manage the Oracle-created enterprise roles. The groups in Active Directory are not visible to either Oracle Identity Management or Fusion Applications.

5.2.3 Setting Up Oracle Internet Directory as a Shadow Directory

In cases where Oracle Internet Directory is used as the shadow directory to store certain attributes, such as all the Fusion Application-specific attributes, use a separate container in Oracle Internet Directory to store the shadow attributes.

- The Shadow Entries container (`cn=shadowentries`) must be in a separate DIT from the parent of the users and groups container `dc=mycompany,dc=com`, as shown in Figure 5-1.
- The same ACL configured for `dc=mycompany,dc=com` within Oracle Internet Directory must be configured for `cn=shadowentries`. To perform this configuration, use the `ldapmodify` command. The syntax is as follows:

```
ldapmodify -D cn=orcladmin -q -p portNum -h hostname -f ldifFile
```

The following is a sample LDIF file to use with `ldapmodify`:

```
dn: cn=shadowentries
changetype: modify
add: orclaci
orclaci: access to entry by
group="cn=RealmAdministrators,cn=groups,cn=OracleContext,dc=mycompany,dc=com"
 (browse,add,delete)
orclaci: access to attr=(*) by
group="cn=RealmAdministrators,cn=groups,cn=OracleContext,dc=mycompany,dc=com"
 (read, write, search, compare)
orclaci: access to entry by
group="cn=OIMAdministrators,cn=groups,dc=mycompany,dc=com"
(browse,add,delete)
orclaci: access to attr = (*) by
group="cn=OIMAdministrators,cn=groups,dc=mycompany,dc=com"
(search,read,compare,write)
-
changetype: modify
add: orclentrylevelaci
orclentrylevelaci: access to entry by * (browse,noadd,nodelete)
orclentrylevelaci: access to attr=(*) by * (read,search,nowrite,nocompare)
```

- If you have more than one directory for which Oracle Internet Directory is used as a Shadow directory, then you must create different shadow containers for each of the directories. The container name can be chosen to uniquely identify the specific directory for which this is a shadow entry.

5.2.4 Directory Structure Overview - Shadow Join

Figure 5-1 shows the directory structure in the primary and shadow directories. The containers `cn=reservation`, `cn=appIDUsers`, `cn=FusionGroups`, and `cn=DataRoleGroups` are speciric to Fusion Applications.

Figure 5-1 Directory Structure

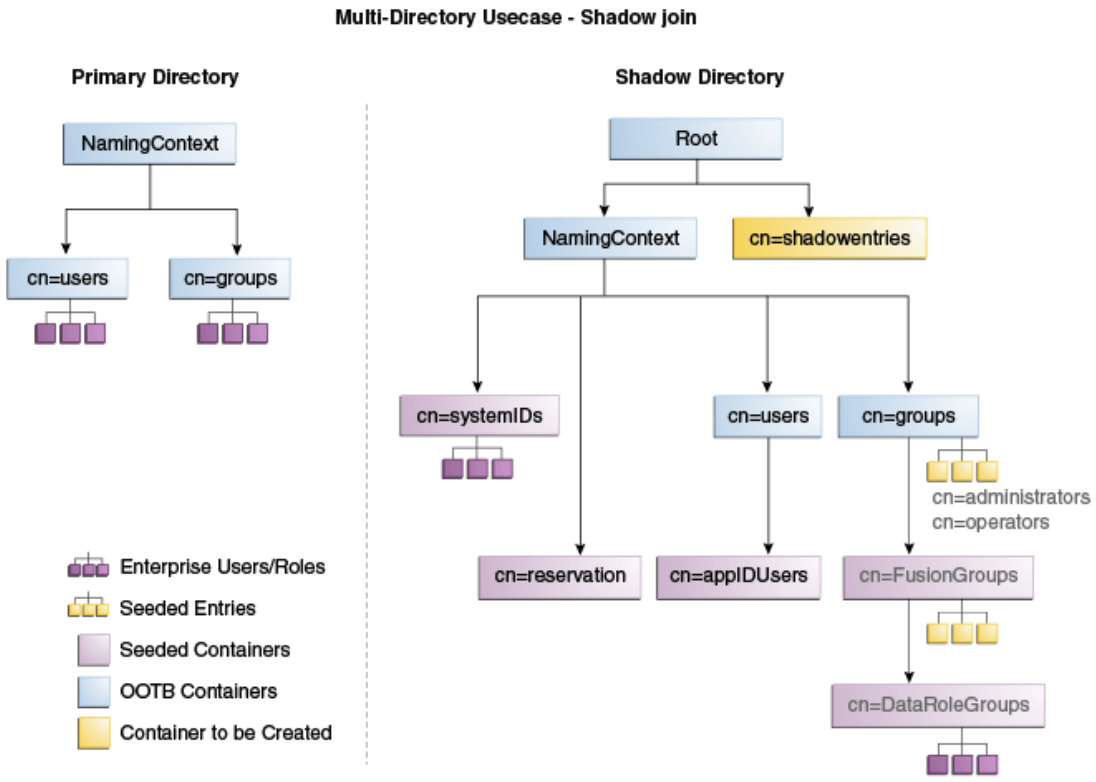


Figure 5-2 shows how the DIT appears to a user or client application. The containers `cn=appIDUsers`, `cn=FusionGroups`, and `cn=DataRoleGroups` are speciric to Fusion Applications.

Figure 5-2 Client View of the DIT

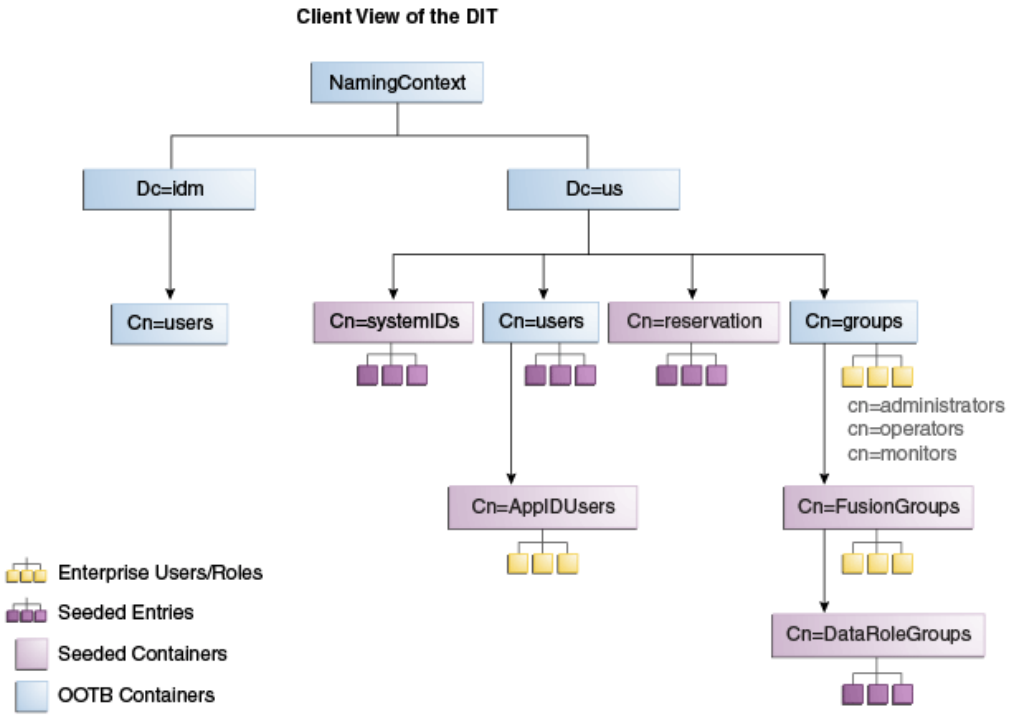
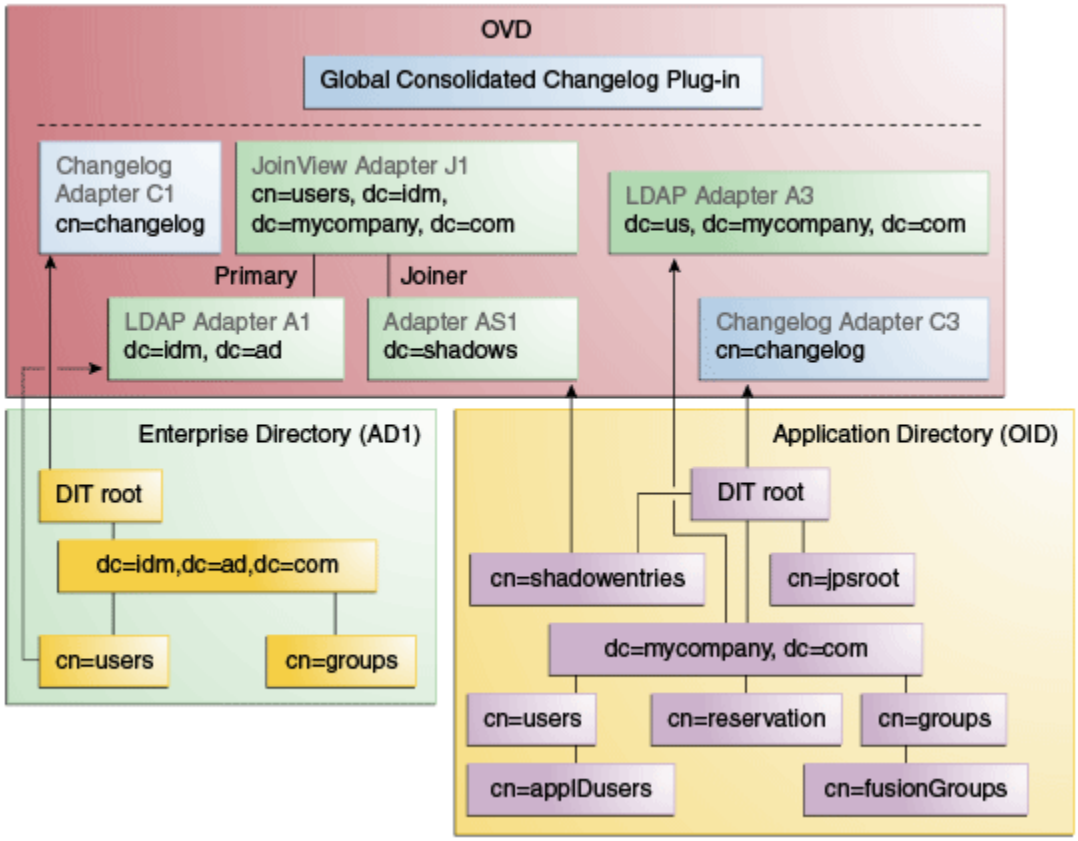


Figure 5-3 summarizes the adapters and plug-ins. The containers `cn=appIDUsers`, and `cn=FusionGroups` are speciric to Fusion Applications.

Figure 5-3 Adapter and Plug-in Configuration



5.2.5 Configuring Oracle Virtual Directory Adapters for Split Profile

In order to produce the client side view of the data shown in Figure 5-2, you must configure multiple adapters in Oracle Virtual Directory following the steps in this section.

You can use `idmConfigTool` to create the adapters to facilitate this configuration.

> **See Also:**
>
> Section A.1, "Verifying Oracle Virtual Directory Adapters for Split Profile by Using ODSM" for instructions on viewing the adapters using Oracle Directory Services Manager.

To create the adapters using `idmConfigTool`, perform the following tasks on IDMHOST1:

1. Set the environment variables: *`MW_HOME`*, *`JAVA_HOME`*, *`IDM_HOME`* and *`ORACLE_HOME`*.

   Set *`IDM_HOME`* to *`IDM_ORACLE_HOME`*

   Set *`ORACLE_HOME`* to *`IAM_ORACLE_HOME`*

2. Create a properties file for the adapter you are configuring called `splitprofile.props`, with the following content:

```
ovd.host:ldaphost1.mycompany.com
ovd.port:8899
ovd.binddn:cn=orcladmin
ovd.ssl:true
ldap1.type:AD
ldap1.host:adhost.mycompany.com
ldap1.port:636
ldap1.binddn:administrator@idmqa.com
ldap1.ssl:true
ldap1.base:dc=idmqa,dc=com
ldap1.ovd.base:dc=idmqa,dc=com
usecase.type:split
ldap2.type:OID
ldap2.host:ldaphost.mycompany.com
ldap2.port:3060
ldap2.binddn:cn=oimLDAP,cn=users,dc=mycompany,dc=com
ldap2.ssl:false
ldap2.base:dc=mycompany,dc=com
ldap2.ovd.base:dc=mycompany,dc=com
```

The following list describes the parameters used in the properties file.

- `ovd.host` is the host name of a server running Oracle Virtual Directory.
- `ovd.port` is the https port used to access Oracle Virtual Directory.
- `ovd.binddn` is the user DN you use to connect to Oracle Virtual Directory.
- `ovd.password` is the password for the DN you use to connect to Oracle Virtual Directory.
- `ovd.oamenabled` is set to `true` if you are using Oracle Access Management Access Manager, otherwise set to `false`.

  `ovd.oamenabled` is always `true` in Fusion Applications deployments.
- `ovd.ssl` is set to `true`, as you are using an https port.
- `ldap1.type` is set to OID for the Oracle Internet Directory back end directory or set to AD for the Active Directory back end directory.
- `ldap1.host` is the Active Directory host. Use the load balancer name where the host is highly available.
- `ldap2.host`: The Oracle Internet Directory host. Use the load balancer name where the host is highly available.
- `ldap1.port` is the port used to communicate with the back end directory.
- `ldap1.binddn` is the bind DN of the `oimLDAP` user.
- `ldap1.password` is the password of the `oimLDAP` user
- `ldap1.ssl` is set to `true` if you are using the back end's SSL connection, and otherwise set to `false`. This should always be set to `true` when an adapter is being created for AD.
- `ldap1.base` is the base location in the directory tree.
- `ldap1.ovd.base` is the mapped location in Oracle Virtual Directory.

- `usecase.type` is set to `Single` when using a single directory type.

3. Configure the adapter by using the `idmConfigTool` command, which is located at:

   IAM_ORACLE_HOME/idmtools/bin

> **Note:**
>
> When you run the `idmConfigTool`, it creates or appends to the file `idmDomainConfig.param`. This file is generated in the same directory that the `idmConfigTool` is run from. To ensure that each time the tool is run, the same file is appended to, always run the `idmConfigTool` from the directory:
>
> *IAM_ORACLE_HOME*/idmtools/bin

The syntax of the command on Linux is:

```
idmConfigTool -configOVD input_file=splitprofile.props
```

During the running of the command you will be prompted for the passwords to each of the directories you will be accessing.

The command must be run once for each Oracle Virtual Directory instance.

5.2.6 Configuring a Global Consolidated Changelog Plug-in

Deploy a global level consolidated changelog plug-in to handle changelog entries from all the Changelog Adapters.

1. In a web browser, go to Oracle Directory Services Manager (ODSM).
2. Connect to an Oracle Virtual Directory instance.
3. On the Home page, click the **Advanced** tab. The Advanced navigation tree appears.
4. Expand **Global Plugins**
5. Click the **Create Plug-In** button. The Plug-In dialog box appears.
6. Enter a name for the Plug-in in the Name field.
7. Select the plug-in class **ConsolidatedChglogPlugin** from the list.
8. Click **OK**.
9. Click **Apply**.

5.2.7 Validating the Oracle Virtual Directory Changelog

Run the following command to validate that the changelog adapter is working:

```
$IDM_ORACLE_HOME/bin/ldapsearch -p 6501 -D cn=orcladmin -q -b 'cn=changelog' -s base
'objectclass=*' lastchangenumber
```

The command should return a changelog result, such as:

```
Please enter bind password:
cn=Changelog
lastChangeNumber=changelog_OID:190048;changelog_AD1:363878
```

If `ldapsearch` does not return a changelog result, double check the changelog adapter configuration.

5.3 Configuring Multiple Directories as an Identity Store: Distinct User and Group Populations in Multiple Directories

In this configuration, you store Oracle-specific entries in Oracle Internet Directory and enterprise-specific entries in Active Directory. If necessary, extend the Active Directory schema as described in "Configuring Active Directory for Use with Oracle Access Management Access Manager and Oracle Identity Manager" in *Oracle® Fusion Middleware Enterprise Deployment Guide for Oracle Identity and Access Management*.

> **Note:**
>
> The Oracle Internet Directory that is to be used is not necessarily the PolicyStore Oracle Internet Directory. Conceptually, a non-Active Directory directory can be used as the second directory. For convenience, this section refers to the Policy Store Oracle Internet Directory.

The following conditions are assumed:

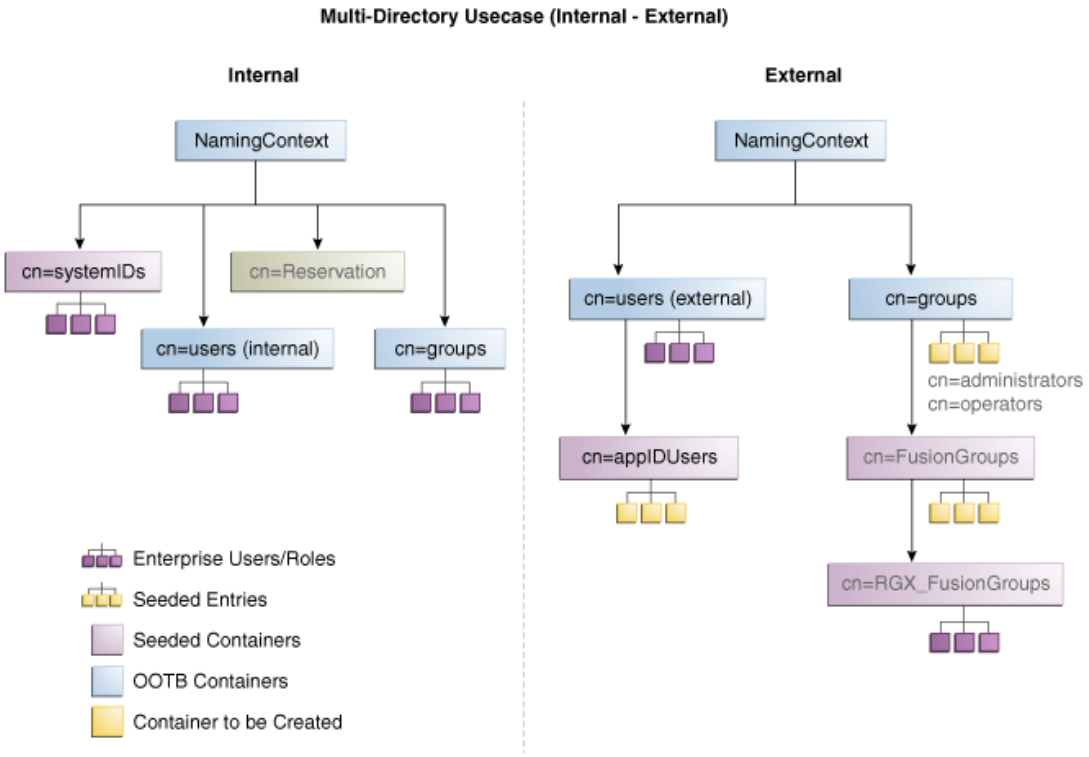
- Enterprise Directory Identity data is in one or more directories. Application-specific attributes of users and groups are stored in the Enterprise Directory.
- Application-specific entries are in the Application Directory. AppIDs and Enterprise Roles are stored in the Application Directory,

This section contains the following topics:

5.3.1 Directory Structure Overview for Distinct User and Group Populations in Multiple Directories

Figure 5-4 shows the directory structure in the two directories, listed here as internal and external. The containers cn=appIDUsers, cn=FusionGroups, and cn=RGX_FusionGroups are Fusion Applications-specific.

Figure 5-4 Directory Structure



Oracle Virtual Directory makes multiple directories look like a single DIT to a user or client application, as shown in Figure 5-5. The containers cn=appIDUsers, cn=FusionGroups, and cn=RGX_FusionGroups are Fusion Applications-specific.

Figure 5-5 Client View of the DIT

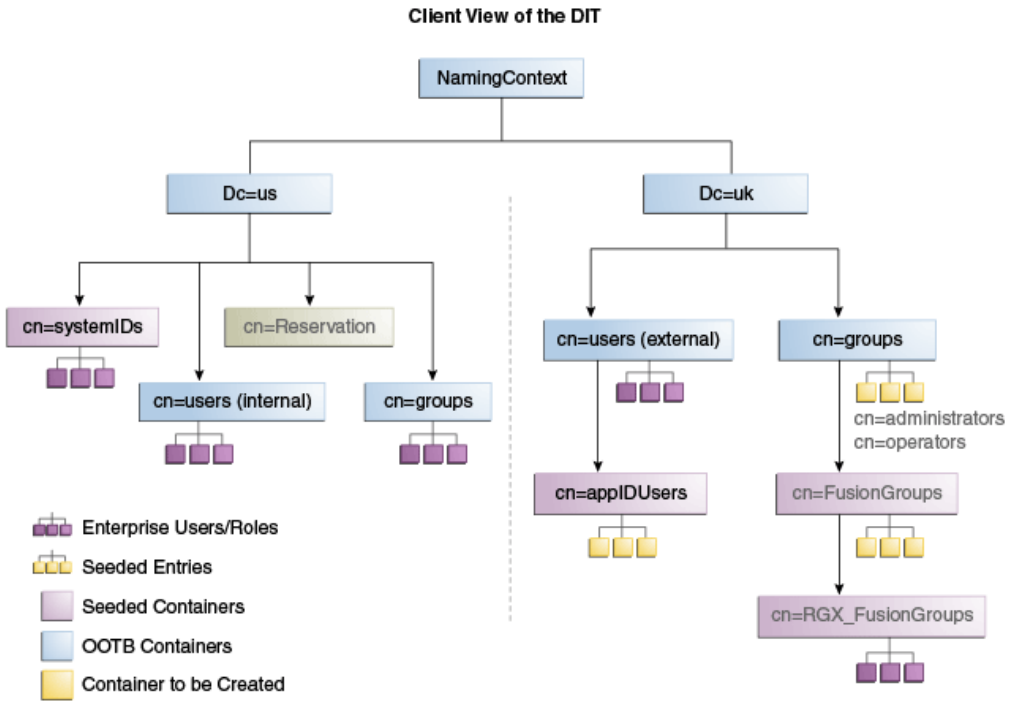
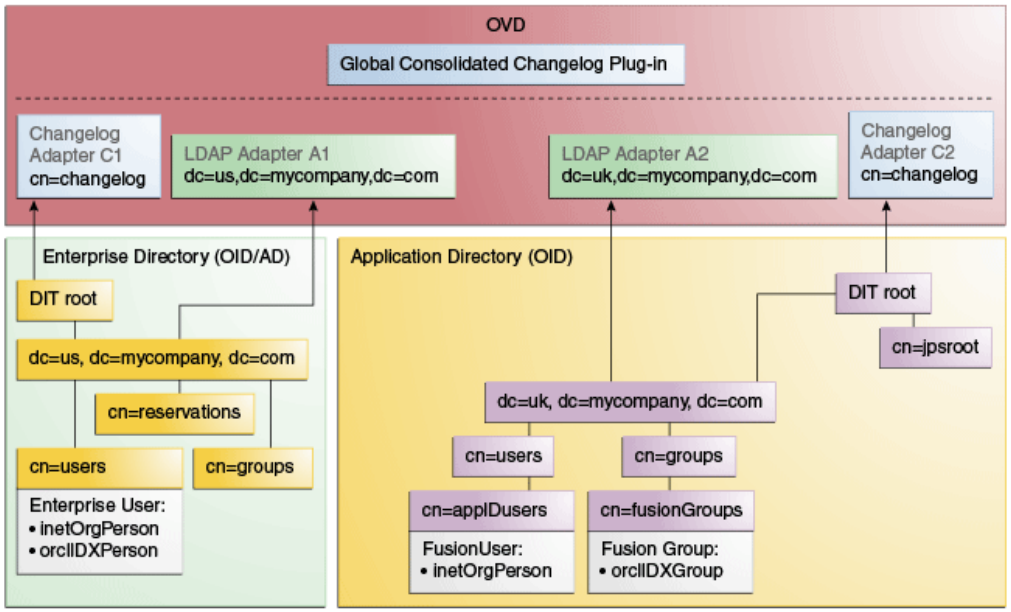


Figure 5-6 provides an overview of the adapter configuration. The classes `inetOrgPerson`, `orclIDXPerson`, and `orclIDXGroup` and the containers `cn=appIDusers` and `cn=fusionGroups` are required only for Fusion Applications.

Figure 5-6 Configuration Overview



5.3.2 Configuring Oracle Virtual Directory Adapters for Distinct User and Group Populations in Multiple Directories

Create the user adapter on the Oracle Virtual Directory instances running on LDAPHOST1 and LDAPHOST2 individually, as described in the following sections:

- Creating Enterprise Directory Adapters
- Creating Application Directory Adapters

5.3.2.1 Creating Enterprise Directory Adapters

Create Oracle Virtual Directory adapters for the Enterprise Directory. The type of adapter that is created will be dependent on whether or not the back end directory resides in Oracle Internet Directory or Active Directory.

You can use `idmconfgTool` to create the Oracle Virtual Directory User and Changelog adapters for Oracle Internet Directory and Active Directory.

> **See Also:**
>
> Section A.1 for instructions on viewing the adapters using Oracle Directory Services Manager.

Oracle Identity Management requires adapters. It is highly recommended, though not mandatory, that you use Oracle Virtual Directory to connect to Oracle Internet Directory.

To create the adapters using `idmconfgTool`, perform the following tasks on IDMHOST1:

1. Set the environment variables: *`MW_HOME`*, *`JAVA_HOME`*, *`IDM_HOME`* and *`ORACLE_HOME`*.

   Set *`IDM_HOME`* to *`IDM_ORACLE_HOME`*

   Set *`ORACLE_HOME`* to *`IAM_ORACLE_HOME`*

2. Create a properties file for the OID or AD adapter you are configuring called `ovd1.props`, as follows:

   > **Note:**
   >
   > The `usecase.type:single` parameter is not supported for Active Directory through the `configOVD` option.

   Oracle Internet Directory adapter properties file:

```
ovd.host:ldaphost1.mycompany.com
ovd.port:8899
ovd.binddn:cn=orcladmin
ovd.password:ovdpassword
ovd.oamenabled:true
ovd.ssl:true
```

```
ldap1.type:OID
ldap1.host:oididstore.mycompany.com
ldap1.port:3060
ldap1.binddn:cn=oimLDAP,cn=systemids,dc=mycompany,dc=com
ldap1.ssl:false
ldap1.base:dc=mycompany,dc=com
ldap1.ovd.base:dc=mycompany,dc=com
usecase.type: single
```

Active Directory adapter properties file:

```
ovd.host:ldaphost1.mycompany.com
ovd.port:8899
ovd.binddn:cn=orcladmin
ovd.password:ovdpassword
ovd.oamenabled:true
ovd.ssl:true
ldap1.type:AD
ldap1.host:adidstore.mycompany.com
ldap1.port:636
ldap1.binddn:cn=adminuser
ldap1.ssl:true
ldap1.base:dc=mycompany,dc=com
ldap1.ovd.base:dc=mycompany,dc=com
usecase.type: single
```

The following list contains the parameters used in the properties file and their descriptions.

- `ovd.host` is the host name of a server running Oracle Virtual Directory.
- `ovd.port` is the https port used to access Oracle Virtual Directory.
- `ovd.binddn` is the user DN you use to connect to Oracle Virtual Directory.
- `ovd.password` is the password for the DN you use to connect to Oracle Virtual Directory.
- `ovd.oamenabled` is set to `true` if you are using Oracle Access Management Access Manager, otherwise set to `false`.

  `ovd.oamenabled` is always `true` in Fusion Applications deployments.
- `ovd.ssl` is set to `true`, as you are using an https port.
- `ldap1.type` is set to OID for the Oracle Internet Directory back end directory or set to AD for the Active Directory back end directory.
- `ldap1.host` Back end directory host.
- `ldap1.port` is the port used to communicate with the back end directory.
- `ldap1.binddn` is the bind DN of the `oimLDAP` user.
- `ldap1.password` is the password of the `oimLDAP` user
- `ldap1.ssl` is set to `true` if you are using the back end's SSL connection, and otherwise set to `false`. This should always be set to `true` when an adapter is being created for AD.
- `ldap1.base` is the base location in the directory tree.
- `ldap1.ovd.base` is the mapped location in Oracle Virtual Directory.
- `usecase.type` is set to `Single` when using a single directory type.

3. Configure the adapter by using the `idmConfigTool` command, which is located at:

   IAM_ORACLE_HOME/idmtools/bin

   > **Note:**
   >
   > When you run the `idmConfigTool`, it creates or appends to the file `idmDomainConfig.param`. This file is generated in the same directory that the `idmConfigTool` is run from. To ensure that each time the tool is run, the same file is appended to, always run the `idmConfigTool` from the directory:
   >
   > *IAM_ORACLE_HOME*/idmtools/bin

   The syntax of the command on Linux is:

   ```
   idmConfigTool.sh -configOVD input_file=configfile [log_file=logfile]
   ```

   The syntax on Windows is:

   ```
   idmConfigTool.bat -configOVD input_file=configfile [log_file=logfile]
   ```

   For example:

   ```
   idmConfigTool.sh -configOVD input_file=ovd1.props
   ```

   The command requires no input. The output looks like this:

   ```
   The tool has completed its operation. Details have been logged to logfile
   ```

Run this command on each Oracle Virtual Directory host in your topology, with the appropriate value for `ovd.host` in the property file.

5.3.2.2 Creating Application Directory Adapters

Create Oracle Virtual Directory adapters for the Application Directory. The back end directory for the application directory is always Oracle Internet Directory.

You can use `idmconfgTool` to create the Oracle Virtual Directory User and Changelog adapters for Oracle Internet Directory and Active Directory. Oracle Identity Management requires adapters. It is highly recommended, though not mandatory, that you use Oracle Virtual Directory to connect to Oracle Internet Directory.

To do this, perform the following tasks on IDMHOST1:

1. Set the environment variables: *MW_HOME*, *JAVA_HOME*, *IDM_HOME* and *ORACLE_HOME*.

   Set *IDM_HOME* to *IDM_ORACLE_HOME*

   Set *ORACLE_HOME* to *IAM_ORACLE_HOME*

2. Create a properties file for the adapter you are configuring called `ovd1.props`. The contents of this file is as follows.

   Oracle Internet Directory adapter properties file:

   ```
   ovd.host:ldaphost1.mycompany.com
   ovd.port:8899
   ovd.binddn:cn=orcladmin
   ovd.password:ovdpassword
   ```

```
ovd.oamenabled:true
ovd.ssl:true
ldap1.type:OID
ldap1.host:oididstore.mycompany.com
ldap1.port:3060
ldap1.binddn:cn=oimLDAP,cn=systemids,dc=mycompany,dc=com
ldap1.password:oidpassword
ldap1.ssl:false
ldap1.base:dc=mycompany,dc=com
ldap1.ovd.base:dc=mycompany,dc=com
usecase.type: single
```

The following list describes the parameters used in the properties file.

- `ovd.host` is the host name of a server running Oracle Virtual Directory.
- `ovd.port` is the https port used to access Oracle Virtual Directory.
- `ovd.binddn` is the user DN you use to connect to Oracle Virtual Directory.
- `ovd.password` is the password for the DN you use to connect to Oracle Virtual Directory.
- `ovd.oamenabled` is set to `true` if you are using Oracle Access Management Access Manager, otherwise set to `false`.

  `ovd.oamenabled` is always `true` in Fusion Applications deployments.
- `ovd.ssl` is set to `true`, as you are using an https port.
- `ldap1.type` is set to OID for the Oracle Internet Directory back end directory or set to AD for the Active Directory back end directory.
- `ldap1.host` is the host on which back end directory is located. Use the load balancer name.
- `ldap1.port` is the port used to communicate with the back end directory.
- `ldap1.binddn` is the bind DN of the `oimLDAP` user.
- `ldap1.password` is the password of the `oimLDAP` user
- `ldap1.ssl` is set to `true` if you are using the back end's SSL connection, and otherwise set to `false`. This should always be set to `true` when an adapter is being created for AD.
- `ldap1.base` is the base location in the directory tree.
- `ldap1.ovd.base` is the mapped location in Oracle Virtual Directory.
- `usecase.type` is set to `Single` when using a single directory type.

3. Configure the adapter by using the `idmConfigTool` command, which is located at:

   IAM_ORACLE_HOME`/idmtools/bin`

> **Note:**
>
> When you run the `idmConfigTool`, it creates or appends to the file `idmDomainConfig.param`. This file is generated in the same directory that the `idmConfigTool` is run from. To ensure that each time the tool is run, the same file is appended to, always run the `idmConfigTool` from the directory:
>
> *IAM_ORACLE_HOME*`/idmtools/bin`

The syntax of the command on Linux is:

```
idmConfigTool.sh -configOVD input_file=configfile [log_file=logfile]
```

The syntax on Windows is:

```
idmConfigTool.bat -configOVD input_file=configfile [log_file=logfile]
```

For example:

```
idmConfigTool.sh -configOVD input_file=ovd1.props
```

The command requires no input. The output looks like this:

```
The tool has completed its operation. Details have been logged to logfile
```

Run this command on each Oracle Virtual Directory host in your topology, with the appropriate value for `ovd.host` in the property file.

5.3.3 Creating a Global Plug-in

To create a Global Oracle Virtual Directory plug-in, proceed as follows:

1. In a web browser, go to Oracle Directory Services Manager (ODSM).
2. Create connections to each of the Oracle Virtual Directory instances running on `LDAPHOST1` and `LDAPHOST2`, if they do not already exist.
3. Connect to each Oracle Virtual Directory instance by using the appropriate connection entry.
4. On the Home page, click the **Advanced** tab. The Advanced navigation tree appears.
5. Click the + next to **Global Plugins** in the left pane.
6. Click **Create Plugin**.
7. Create the Global Consolidated Changelog Plug-in as follows:

   Enter the following values to create the Global Consolidated Plug-in:

   - **Name**: `Global Consolidated Changelog`
   - **Class**: Click **Select** then choose: **ConsolidatedChangelog**

   Click **OK** when finished.

The environment is now ready to be configured to work with Oracle Virtual Directory as the Identity Store.

5.4 Additional Configuration Tasks When Reintegrating Oracle Identity Governance With Multiple Directories

If you have previously integrated Oracle Identity Management with a single directory and you are now reintegrating it with multiple directories, you must reset the changelog number for each of the incremental jobs to zero. The changelog numbers are repopulated on the next run.

Part V
Appendices

This part contains supplementary content to support the procedures in the book, and includes the following appendices:

- Verifying Adapters for Multiple Directory Identity Stores by Using ODSM
- Using the idm.conf File
- Enabling LDAP Synchronization in Oracle Identity Governance
- Configuring Oracle Virtual Directory for Integration with Oracle Access Management Access Manager

A

Verifying Adapters for Multiple Directory Identity Stores by Using ODSM

After you have configured your Oracle Virtual Directory adapters as described in Chapter 6, "Configuring an Identity Store with Multiple Directories," you can use ODSM to view the adapters for troubleshooting purposes. This chapter explains how.
This appendix contains the following sections:

- Verifying Oracle Virtual Directory Adapters for Split Profile by Using ODSM
- Verifying Adapters for Distinct User and Group Populations in Multiple Directories by Using ODSM

A.1 Verifying Oracle Virtual Directory Adapters for Split Profile by Using ODSM

This section describes how to validate the adapters created in Configuring Oracle Virtual Directory Adapters for Split Profile.

This section contains the following topics:

- Verifying User Adapter for Active Directory Server
- Verifying Shadowjoiner User Adapter
- Verifying JoinView Adapter
- Verifying User/Role Adapter for Oracle Internet Directory
- Verifying Changelog Adapter for Active Directory Server
- Verifying Changelog Adapter for Oracle Internet Directory
- Configuring a Global Consolidated Changelog Plug-in
- Validating Oracle Virtual Directory Changelog

A.1.1 Verifying User Adapter for Active Directory Server

Verify the following adapter and plug-ins for Active Directory:

Follow these steps to verify the User Adapter in Oracle Virtual Directory using Oracle Directory Services Manager.

1. In a web browser, go to Oracle Directory Services Manager (ODSM). The URL is of the form: `http://admin.mycompany.com/odsm`.
2. Connect to each Oracle Virtual Directory instance by using the appropriate connection entry.
3. On the Home page, click the **Adapter** tab.
4. Click **user_AD1** adapter.

5. Verify that the User Adapter routing as configured correctly:
   a. **Visibility** must be set to internal.
   b. **Bind Support** must be set to enable.
6. Verify the User Adapter User Management Plug-in as follows:
   a. Select the **User Adapter**.
   b. Click the **Plug-ins** tab.
   c. Click the **User Management** Plug-in, then click **Edit** in the plug-ins table. The plug-in editing window appears.
   d. Verify that the plug-in parameters are as follows:

Parameter	Value	Default
directoryType	`activedirectory`	`Yes`
exclusionMapping	`orclappiduser,uid=samaccountname`	
mapAttribute	`orclguid=objectGuid`	
mapAttribute	`uniquemember=member`	
addAttribute	`user,samaccountname=%uid%,%orclshortuid%`	
mapAttribute	`mail=userPrincipalName`	
mapAttribute	`ntgrouptype=grouptype`	
mapObjectclass	`groupofUniqueNames=group`	
mapObjectclass	`orclidxperson=user`	
pwdMaxFailure	10	`Yes`
oamEnabled	`True`[1]	
mapObjectClass	`inetorgperson=user`	`Yes`
mapPassword	`True`	`Yes`
oimLanguages	Comma separated list of language codes, such as `en,fr,ja`	

[1] Set oamEnabled to true only if you are using Oracle Access Management Access Manager.

A.1.2 Verifying Shadowjoiner User Adapter

Follow these steps to verify the ShadowJoiner Adapter in Oracle Virtual Directory using Oracle Directory Services Manager.

1. In a web browser, go to Oracle Directory Services Manager (ODSM).
2. Connect to Oracle Virtual Directory.
3. On the Home page, click the **Adapter** tab.
4. Click the **Shadow4AD1** Adapter.
5. Ensure that User Adapter routing as is configured correctly:
   a. **Visibility** must be set to internal.

b. **Bind Support** must be set to enable.

6. Verify the User Adapter as follows:
   a. Select the User Adapter.
   b. Click the **Plug-ins** tab.
   c. Click the **User Management** Plug-in, then click **Edit** in the plug-ins table. The plug-in editing window appears.
   d. Verify that the parameters are as follows:

Parameter	Value	Default
directoryType	`oid`	`Yes`
pwdMaxFailure	`10`	`Yes`
oamEnabled	`true`	
mapObjectclass	`container=orclContainer`	`Yes`
oimDateFormat	yyyyMMddHHmmss'z'	

A.1.3 Verifying JoinView Adapter

Follow these steps to verify the User Adapter in Oracle Virtual Directory using Oracle Directory Services Manager.

1. In a web browser, go to the Oracle Directory Services Manager (ODSM) page.
2. Connect to Oracle Virtual Directory.
3. On the Home page, click the **Adapter** tab.
4. Click the JoinView adapter.
5. Verify the Adapter as follows
   a. Click **Joined Adapter** in the adapter tree. It should exist
   b. Click **OK**.

A.1.4 Verifying User/Role Adapter for Oracle Internet Directory

Follow these steps to verify the User Adapter in Oracle Virtual Directory using Oracle Directory Services Manager.

1. In a web browser, go to Oracle Directory Services Manager (ODSM).
2. Connect to Oracle Virtual Directory.
3. On the Home page, click the **Adapter** tab.
4. Click User Adapter.
5. Verify the plug-in as follows:
   a. Select the User Adapter.
   b. Click the **Plug-ins** tab.
   c. Click the **User Management** Plug-in in the plug-ins table, then click **Edit**. The plug-in editing window appears.

d. Verify that the parameters are as follows:

Parameter	Value	Default
directoryType	`oid`	`Yes`
pwdMaxFailure	`10`	`Yes`
oamEnabled	`true`	
mapObjectclass	`container=orclContainer`	`Yes`
oimDateFormat	yyyyMMddHHmmss'z'	

e. Click **OK**.

A.1.5 Verifying Changelog Adapter for Active Directory Server

Follow these steps to verify the Changelog Adapter in Oracle Virtual Directory using Oracle Directory Services Manager.

1. In a web browser, go to Oracle Directory Services Manager (ODSM).
2. Connect to Oracle Virtual Directory.
3. On the Home page, click the **Adapter** tab.
4. Click the changelog_AD1 adapter.
5. Verify the plug-in as follows.
   a. Select the Changelog Adapter.
   b. Click the **Plug-ins** tab.
   c. In the Deployed Plus-ins table, click the **changelog** plug-in, then click "**Edit** in the plug-ins table. The plug-in editing window appears.
   d. Verify that the parameter values are as follows:

Parameter	Value
directoryType	`activedirectory`
mapAttribute	`targetGUID=objectGUID`
requiredAttribute	`samaccountname`
sizeLimit	`1000`
targetDNFilter	`cn=users,dc=idm,dc=ad,dc=com` The users container in Active Directory
mapUserState	`true`
oamEnabled	`true`
virtualDITAdapterName	user_J1;user_AD1

A.1.6 Verifying Changelog Adapter for Oracle Internet Directory

To use the changelog adapter, you must first enable changelog on the connected directory. To test whether the directory is changelog enabled, type:

```
ldapsearch -h directory_host -p ldap_port -D bind_dn -q -b '' -s base 'objectclass=*'
lastchangenumber
```

for example:

```
ldapsearch -h ldaphost1 -p 389 -D "cn=orcladmin" -q -b '' -s base 'objectclass=*'
lastchangenumber
```

If you see `lastchangenumber` with a value, it is enabled. If it is not enabled, enable it as described in the Enabling and Disabling Changelog Generation by Using the Command Line section of *Administering Oracle Internet Directory*.

Follow these steps to verify the Changelog Adapter in Oracle Virtual Directory using Oracle Directory Services Manager.

1. In a web browser, go to Oracle Directory Services Manager (ODSM).
2. Connect to an Oracle Virtual Directory instance.
3. On the Home page, click the **Adapter** tab.
4. Click the Changelog Adapter.
5. Verify the plug-in as follow.
   a. Select the Changelog Adapter.
   b. Click the **Plug-ins** tab.
   c. In the Deployed Plug-ins table, click the **changelog** plug-in, then click **Edit** in the plug-ins table. The plug-in editing window appears.
   d. Verify that the parameter values are as follows:

Parameter	Value
directoryType	`oid`
mapAttribute	`targetGUID=orclguid`
requiredAttribute	`orclGUID`
modifierDNFilter	`cn=orcladmin`
sizeLimit	`1000`
targetDNFilter	`dc=mycompany,dc=com`
targetDNFilter	cn=shadowentries
mapUserState	`true`
oamEnabled	`true`
virtualDITAdapterName	user_J1;shadow4AD1
virtualDITAdapterName	User Adapter (The name of the User adapter's name)

A.1.7 Configuring a Global Consolidated Changelog Plug-in

Verify the global level consolidated changelog plug-in as follows

1. In a web browser, go to Oracle Directory Services Manager (ODSM).
2. Connect to an Oracle Virtual Directory instance.
3. On the Home page, click the **Advanced** tab. The Advanced navigation tree appears.

4. Expand **Global Plugins**
5. Click the **ConsolidatedChglogPlugin**. The plug-in editing window appears.

A.1.8 Validating Oracle Virtual Directory Changelog

Run the following command to validate that the changelog adapter is working:

```
$IDM_ORACLE_HOME/bin/ldapsearch -p 6501 -D cn=orcladmin -q -b 'cn=changelog' -s
base 'objectclass=*' lastchangenumber
```

The command should return a changelog result, such as:

```
Please enter bind password:
cn=Changelog
lastChangeNumber=changelog_OID:190048;changelog_AD1:363878
```

If `ldapsearch` does not return a changelog result, double check the changelog adapter configuration.

A.2 Verifying Adapters for Distinct User and Group Populations in Multiple Directories by Using ODSM

This section describes how to view the adapters created in Configuring Oracle Virtual Directory Adapters for Distinct User and Group Populations in Multiple Directories.

This section contains the following topics:

- Verifying the User Adapter on the Oracle Virtual Directory Instances
- Verifying the Plug-In of the User/Role Adapter A1
- Verifying the Plug-In of the User/Role Adapter A2
- Verifying the Changelog Adapter C1 Plug-In
- Verifying the Changelog Adapter for Active Directory
- Verifying Changelog Adapter C2
- Verifying Oracle Virtual Directory Global Plug-in
- Configuring a Global Consolidated Changelog Plug-in

A.2.1 Verifying the User Adapter on the Oracle Virtual Directory Instances

Verify the user adapter on the Oracle Virtual Directory instances running on LDAPHOST1 and LDAPHOST2 individually. Follow these steps to verify the User Adapter in Oracle Virtual Directory using Oracle Directory Services Manager:

1. If they are not already running, start the Administration Server and the WLS_ODSM Managed Servers.
2. In a web browser, go to Oracle Directory Services Manager (ODSM) at:

   `http://admin.mycompany.com/odsm`

3. Verify connections to each of the Oracle Virtual Directory instances running on LDAPHOST1 and LDAPHOST2, if they do not already exist.
4. Connect to each Oracle Virtual Directory instance by using the appropriate connection entry.
5. On the Home page, click the **Adapter** tab.
6. Click the name of each adapter. Verify that it has the parameters shown in the following tables.

A.2.2 Verifying the Plug-In of the User/Role Adapter A1

Verify the plug-in of the User/Role Adapter A1, as follows:

1. Select the OIM User Adapter.
2. Click the **Plug-ins** tab.
3. Click the **User Management** Plug-in, then click **Edit** in the plug-ins table. The plug-in editing window appears.
4. Verify that the parameter values are as follows:

Parameter	Value	Default
directoryType	`activedirectory`	`Yes`
exclusionMapping	`orclappiduser,uid=samaccountname`	
mapAttribute	`orclguid=objectGuid`	
mapAttribute	`uniquemember=member`	
addAttribute	`user,samaccountname=%uid%,%orclshortuid%`	
mapAttribute	`mail=userPrincipalName`	
mapAttribute	`ntgrouptype=grouptype`	
mapObjectclass	`groupofUniqueNames=group`	
mapObjectclass	`orclidxperson=user`	
pwdMaxFailure	10	`Yes`
oamEnabled	`True`[1]	
mapObjectClass	`inetorgperson=user`	`Yes`
mapPassword	`True`	`Yes`
oimLanguages	Comma separated list of language codes, such as `en,fr,ja`	

[1] Set oamEnabled to true only if you are using Oracle Access Management Access Manager.

A.2.3 Verifying the Plug-In of the User/Role Adapter A2

Verify the plug-in of the User/Role Adapter A2 as follows:

1. Select the User Adapter.
2. Click the **Plug-ins** tab.

3. Click the **User Management** Plug-in in the plug-ins table, then click **Edit**. The plug-in editing window appears.
4. Verify that the parameter values are as follows:

Parameter	Value	Default
directoryType	`oid`	`Yes`
pwdMaxFailure	`10`	`Yes`
oamEnabled	`true`[1]	
mapObjectclass	`container=orclContainer`	`Yes`

[1] Set oamEnabled to true only if you are using Oracle Access Management Access Manager.

A.2.4 Verifying the Changelog Adapter C1 Plug-In

To verify the Changelog Adapter C1 plug-in, follow these steps:

1. Select the OIM changelog adapter **Changelog_Adapter_C1**.
2. Click the **Plug-ins** tab.
3. In the **Deployed Plus-ins** table, click the **changelog** plug-in, then click **Edit** in the plug-ins table. The plug-in editing window appears.
4. In the **Parameters** table, verify that the values are as shown.

Table A-1 Values in Parameters Table

Parameter	Value	Comments
modifierDNFilter	A bind DN that has administrative rights on the directory server, in the format: "!(modifiersname=cn=*BindDN*)" For example: "!(modifiersname=cn=orcladmin,cn=systemids,dc=mycompany,dc=com)"	Create
sizeLimit	1000	Create
targetDNFilter	`dc=us,dc=mycompany,dc=com`	Create
mapUserState	true	Update
oamEnabled	true	Update
virtualDITAdapterName	The adapter name of User/Role Adapter A1: `User_Adapter_A1`	Create

A.2.5 Verifying the Changelog Adapter for Active Directory

Verify the plug-in as follows.

1. Select the OIM Changelog Adapter.
2. Click the **Plug-ins** tab.

3. In the Deployed Plus-ins table, click the **changelog** plug-in, then click "**Edit** in the plug-ins table. The plug-in editing window appears.
4. In the Parameters table, verify that the parameters are as follows:

Parameter	Value
directoryType	`activedirectory`
mapAttribute	`targetGUID=objectGUID`
requiredAttribute	`samaccountname`
sizeLimit	`1000`
targetDNFilter	`dc=mycompany,dc=com` Search base from which reconciliation must happen. This value must be the same as the LDAP SearchDN that is specified during Oracle Identity Manager installation.
mapUserState	`true`
oamEnabled	`true`[1]
virtualDITAdapterName	The name of the User adapter's name

[1] Set oamEnabled to true only if you are using Oracle Access Management Access Manager.

> **Note:**
>
> **virtualDITAdapterName** identifies the corresponding user profile adapter name. For example, in a single-directory deployment, you can set this parameter value to `User Adapter`, which is the user adapter name. In a split-user profile scenario, you can set this parameter to `J1;A2`, where `J1` is the JoinView adapter name, and `A2` is the corresponding user adapter in the `J1`.

A.2.6 Verifying Changelog Adapter C2

Verify the plug-in as follows:

1. Select the OIM changelog adapter **Changelog_Adapter_C2**.
2. Click the **Plug-ins** tab.
3. In the **Deployed Plus-ins** table, click the **changelog** plug-in, then click **Edit** in the plug-ins table. The plug-in editing window appears.
4. In the **Parameters** table, verify that the parameters are as follows:

Table A-2 Values in Parameters Table

Parameter	Value	Comments
modifierDNFilter	A bind DN that has administrative rights on the directory server, in the format: "!(modifiersname=cn=*BindDN*)" For example: "!(modifiersname=cn=orcladmin,dc=mycompany,dc=com)"	Create
sizeLimit	1000	Create
targetDNFilter	dc=uk,dc=mycompany,dc=com	Create
mapUserState	true	Update
oamEnabled	true	Update
virtualDITAdapterName	The adapter name of User/Role adapter A2: User_Adapter_A2	Create

A.2.7 Verifying Oracle Virtual Directory Global Plug-in

To verify the Global Oracle Virtual Directory plug-in, proceed as follows

1. In a web browser, go to Oracle Directory Services Manager (ODSM) at:

   `http://admin.mycompany.com/odsm`

2. Verify connections to each of the Oracle Virtual Directory instances running on LDAPHOST1 and LDAPHOST2, if they do not already exist.
3. Connect to each Oracle Virtual Directory instance by using the appropriate connection entry.
4. On the Home page, click the **Adapter** tab.
5. Click the **Plug-ins** tab.
6. Verify that the Global Consolidated Changelog Plug-in exists.

   Click **OK** when finished.

A.2.8 Configuring a Global Consolidated Changelog Plug-in

Verify the global level consolidated changelog plug-in as follows

1. In a web browser, go to Oracle Directory Services Manager (ODSM).
2. Connect to an Oracle Virtual Directory instance.
3. On the Home page, click the **Advanced** tab. The Advanced navigation tree appears.
4. Expand **Global Plugins**
5. Click the **ConsolidatedChglogPlugin**. The plug-in editing window appears.

B
Using the idm.conf File

This appendix explains the purpose and usage of the idm.conf file for applications with a web interface.
This appendix contains the following topics:

- About the idm.conf File
- Example idm.conf File

B.1 About the idm.conf File

In the Oracle Fusion Middleware environment, the highest level configuration file at the web tier is httpd.conf. This file configures OHS, which processes the web transactions that use the http protocol. OHS processes each incoming request and determines its routing based on the URL from which the request originates and the resource to be accessed.

Additional configuration files are specified in the httpd.conf file by means of the Apache HTTP Server's `Include` directive in an `Ifmodule` block.

Identity management applications in particular make use of the `idm.conf` configuration file, which is a template that administrators can modify to indicate how incoming requests for protected applications must be handled.

The `idm.conf` configuration file is divided into four parts, each addressing a distinct security area or zone. Table B-1 lists the zones:

Table B-1 Zones in the idm.conf File

Zone	Type	Description
1	Default Access Zone	This zone is the default OHS endpoint for all inbound traffic. The protocol is `http` and the context root is in the format `authohs.example.com:7777`.
2	External Access Zone	This zone is the load-balancer (LBR) external end user endpoint. The protocol is `https` and the context root is in the format `sso.example.com:443`.
3	Internal Services Zone	This zone is the LBR internal endpoint for applications. The protocol is `http` and the context root is in the format `idminternal.example.com:7777`.
4	Administrative Services Zone	This zone is the LBR internal endpoint for administrative services. The protocol is `https` and the context root is in the format `admin.example.com:443`.

When updating the `idm.conf` file, be sure to edit only the zone definition applicable to your requirements.

B.2 Example idm.conf File

The following sample shows the layout and different zones of the idm.conf file:

```
NameVirtualHost *:7777

## Default Access
## AUTHOHS.EXAMPLE.COM

<VirtualHost *:7777>
#  ServerName http://authohs.example.com:7777 (replace the ServerName below with
the actual host:port)
   ServerName http://authohs.us.example.com:7777
   RewriteEngine On
   RewriteRule ^/console/jsp/common/logout.jsp "/oamsso/logout.html?end_url=/
console" [R]
   RewriteRule ^/em/targetauth/emaslogout.jsp "/oamsso/logout.html?end_url=/em"
[R]
   RewriteRule ^/FSMIdentity/faces/pages/Self.jspx "/oim" [R]
   RewriteRule ^/FSMIdentity/faces/pages/pwdmgmt.jspx "/admin/faces/pages/
pwdmgmt.jspx" [R]
   RewriteOptions inherit
   UseCanonicalName On

# Admin Server and EM

   <Location /console>
      SetHandler weblogic-handler
      WebLogicHost us.example.com
      WeblogicPort 17001
   </Location>

   <Location /consolehelp>
      SetHandler weblogic-handler
      WebLogicHost us.example.com
      WeblogicPort 17001
   </Location>

   <Location /em>
      SetHandler weblogic-handler
      WebLogicHost us.example.com
      WeblogicPort 17001
   </Location>

# FA service

   <Location /fusion_apps>
      SetHandler weblogic-handler
      WebLogicHost us.example.com
      WebLogicPort 14100
   </Location>

#ODSM Related entries
   <Location /odsm>
        SetHandler weblogic-handler
        WLProxySSL ON
        WLProxySSLPassThrough ON
        WebLogicHost oidfa.us.example.com
        WeblogicPort 7005
```

```
   </Location>

# OAM Related Entries

   <Location /oamconsole>
      SetHandler weblogic-handler
      WebLogicHost us.example.com
      WebLogicPort 17001
   </Location>

   <Location /oam>
      SetHandler weblogic-handler
      WebLogicHost us.example.com
      WebLogicPort 14100
   </Location>

# OIM Related Entries

# oim identity self service console
<Location /identity>
   SetHandler weblogic-handler
   WLProxySSL ON
   WLProxySSLPassThrough ON
   WLCookieName oimjsessionid
   WebLogicHost us.example.com

    WeblogicPort 14000
 WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>
# oim identity system administration console
  <Location /sysadmin>
     SetHandler weblogic-handler
     WLProxySSL ON
     WLProxySSLPassThrough ON
     WLCookieName oimjsessionid
     WebLogicHost us.example.com
     WeblogicPort 14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>
# oim identity advanced administration console - Legacy 11gR1 webapp
  <Location /oim>
     SetHandler weblogic-handler
     WLProxySSL ON
     WLProxySSLPassThrough ON
     WLCookieName oimjsessionid
     WebLogicHost us.example.com
     WeblogicPort 14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# xlWebApp - Legacy 9.x webapp (struts based)
   <Location /xlWebApp>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# Nexaweb WebApp - used for workflow designer and DM
```

```
   <Location /Nexaweb>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# spml xsd profile
   <Location /spml-xsd>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# used for FA Callback service.
   <Location /callbackResponseService>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# Role-SOD profile
   <Location /role-sod>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# SOA Callback webservice for SOD - Provide the SOA Managed Server Ports
   <Location /sodcheck>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 8001

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# Callback webservice for SOA. SOA calls this when a request is approved/rejected
# Provide the SOA Managed Server Port
   <Location /workflowservice>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# HTTP client service
```

```
   <Location /HTTPClnt>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>


# OIF Related Entries

   <Location /fed>
      SetHandler weblogic-handler
      WebLogicHost us.example.com
      WebLogicPort 7499
   </Location>

</VirtualHost>

## External Access
## SSO.EXAMPLE.COM

<VirtualHost *:7777>
#  ServerName https://sso.example.com:443 (replace the ServerName below with the
actual host:port)
   ServerName https://sso.example.com:443
   RewriteEngine On
   RewriteRule ^/console/jsp/common/logout.jsp "/oamsso/logout.html?end_url=/console"
[R]
   RewriteRule ^/em/targetauth/emaslogout.jsp "/oamsso/logout.html?end_url=/em" [R]
   RewriteRule ^/FSMIdentity/faces/pages/Self.jspx "/oim" [R]
   RewriteRule ^/FSMIdentity/faces/pages/pwdmgmt.jspx "/admin/faces/pages/
pwdmgmt.jspx" [R]
   RewriteOptions inherit
   UseCanonicalName On

# FA service
   <Location /fusion_apps>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WebLogicHost us.example.com
      WebLogicPort 14100
   </Location>

# OAM Related Entries

   <Location /oam>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WebLogicHost us.example.com
      WebLogicPort 14100
   </Location>

# OIM Related Entries

# oim identity self service console
<Location /identity>
   SetHandler weblogic-handler
```

```
   WLProxySSL ON
   WLProxySSLPassThrough ON
   WLCookieName oimjsessionid
   WebLogicHost us.example.com

          WeblogicPort 14000
 WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>
# oim identity system administration console
  <Location /sysadmin>
     SetHandler weblogic-handler
     WLProxySSL ON
     WLProxySSLPassThrough ON
     WLCookieName oimjsessionid
     WebLogicHost us.example.com
     WeblogicPort 14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>
# oim identity advanced administration console - Legacy 11gR1 webapp
  <Location /oim>
     SetHandler weblogic-handler
     WLProxySSL ON
     WLProxySSLPassThrough ON
     WLCookieName oimjsessionid
     WebLogicHost us.example.com
     WeblogicPort 14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# xlWebApp - Legacy 9.x webapp (struts based)
   <Location /xlWebApp>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# Nexaweb WebApp - used for workflow designer and DM
   <Location /Nexaweb>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# spml xsd profile
   <Location /spml-xsd>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000
```

```
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# used for FA Callback service.
   <Location /callbackResponseService>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# OIF Related Entries
   <Location /fed>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WebLogicHost weblogic-host.example.com
      WebLogicPort 7499
   </Location>

</VirtualHost>

## IDM Internal services for FA
## IDMINTERNAL.EXAMPLE.COM

<VirtualHost *:7777>
#  ServerName http://idminternal.example.com:7777 (replace the ServerName below with
the actual host:port)
   ServerName http://idminternal.example.com:7777
   RewriteEngine On
   RewriteRule ^/console/jsp/common/logout.jsp "/oamsso/logout.html?end_url=/console"
[R]
   RewriteRule ^/em/targetauth/emaslogout.jsp "/oamsso/logout.html?end_url=/em" [R]
   RewriteRule ^/FSMIdentity/faces/pages/Self.jspx "/oim" [R]
   RewriteRule ^/FSMIdentity/faces/pages/pwdmgmt.jspx "/admin/faces/pages/
pwdmgmt.jspx" [R]
   RewriteOptions inherit
   UseCanonicalName On

# FA service
   <Location /fusion_apps>
      SetHandler weblogic-handler
      WebLogicHost us.example.com
      WebLogicPort 14100
   </Location>


# OAM Related Entries

   <Location /oam>
      SetHandler weblogic-handler
      WebLogicHost us.example.com
      WebLogicPort 14100
   </Location>

# OIM Related Entries
```

```
# oim identity self service console
<Location /identity>
   SetHandler weblogic-handler
   WLProxySSL ON
   WLProxySSLPassThrough ON
   WLCookieName oimjsessionid
   WebLogicHost us.example.com

          WeblogicPort 14000
 WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>
# oim identity system administration console
  <Location /sysadmin>
     SetHandler weblogic-handler
     WLProxySSL ON
     WLProxySSLPassThrough ON
     WLCookieName oimjsessionid
     WebLogicHost us.example.com
     WeblogicPort 14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>
# oim identity advanced administration console - Legacy 11gR1 webapp
  <Location /oim>
     SetHandler weblogic-handler
     WLProxySSL ON
     WLProxySSLPassThrough ON
     WLCookieName oimjsessionid
     WebLogicHost us.example.com
     WeblogicPort 14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# xlWebApp - Legacy 9.x webapp (struts based)
   <Location /xlWebApp>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# Nexaweb WebApp - used for workflow designer and DM
   <Location /Nexaweb>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# spml xsd profile
   <Location /spml-xsd>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
```

```
   </Location>

# used for FA Callback service.
   <Location /callbackResponseService>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# Role-SOD profile
   <Location /role-sod>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# SOA Callback webservice for SOD - Provide the SOA Managed Server Ports
   <Location /sodcheck>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 8001

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# Callback webservice for SOA. SOA calls this when a request is approved/rejected
# Provide the SOA Managed Server Port
   <Location /workflowservice>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# HTTP client service
   <Location /HTTPClnt>
      SetHandler weblogic-handler
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# OIF Related Entries

   <Location /fed>
      SetHandler weblogic-handler
      WebLogicHost us.example.com
      WebLogicPort 7499
   </Location>
```

```
</VirtualHost>

## IDM Admin services for FA
## ADMIN.EXAMPLE.COM

<VirtualHost *:7777>
#  ServerName https://admin.example.com:443 (replace the ServerName below with
the actual host:port)
   ServerName https://admin.example.com:443
   RewriteEngine On
   RewriteRule ^/console/jsp/common/logout.jsp "/oamsso/logout.html?end_url=/
console" [R]
   RewriteRule ^/em/targetauth/emaslogout.jsp "/oamsso/logout.html?end_url=/em"
[R]
   RewriteRule ^/FSMIdentity/faces/pages/Self.jspx "/oim" [R]
   RewriteRule ^/FSMIdentity/faces/pages/pwdmgmt.jspx "/admin/faces/pages/
pwdmgmt.jspx" [R]
   RewriteOptions inherit
   UseCanonicalName On

# Admin Server and EM

   <Location /console>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WebLogicHost us.example.com
      WeblogicPort 17001
   </Location>

   <Location /consolehelp>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WebLogicHost us.example.com
      WeblogicPort 17001
   </Location>

   <Location /em>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WebLogicHost us.example.com
      WeblogicPort 17001
   </Location>

#ODSM Related entries
   <Location /odsm>
        SetHandler weblogic-handler
        WLProxySSL ON
        WLProxySSLPassThrough ON
        WebLogicHost oidfa.us.example.com
        WeblogicPort 7005
   </Location>

# OAM Related Entries

   <Location /oamconsole>
      SetHandler weblogic-handler
      WLProxySSL ON
```

```
      WLProxySSLPassThrough ON
      WebLogicHost us.example.com
      WebLogicPort 17001
   </Location>


# OIM Related Entries

# oim identity self service console
<Location /identity>
   SetHandler weblogic-handler
   WLProxySSL ON
   WLProxySSLPassThrough ON
   WLCookieName oimjsessionid
   WebLogicHost us.example.com

    WeblogicPort 14000
 WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>
# oim identity system administration console
  <Location /sysadmin>
     SetHandler weblogic-handler
     WLProxySSL ON
     WLProxySSLPassThrough ON
     WLCookieName oimjsessionid
     WebLogicHost us.example.com
     WeblogicPort 14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>
# oim identity advanced administration console - Legacy 11gR1 webapp
  <Location /oim>
     SetHandler weblogic-handler
     WLProxySSL ON
     WLProxySSLPassThrough ON
     WLCookieName oimjsessionid
     WebLogicHost us.example.com
     WeblogicPort 14000
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>


# xlWebApp - Legacy 9.x webapp (struts based)
   <Location /xlWebApp>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# Nexaweb WebApp - used for workflow designer and DM
   <Location /Nexaweb>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000
```

```
    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# HTTP client service
   <Location /HTTPClnt>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WLCookieName oimjsessionid
      WebLogicHost us.example.com
      WeblogicPort 14000

    WLLogFile "${ORACLE_INSTANCE}/diagnostics/logs/mod_wl/oim_component.log"
   </Location>

# OIF Related Entries
   <Location /fed>
      SetHandler weblogic-handler
      WLProxySSL ON
      WLProxySSLPassThrough ON
      WebLogicHost weblogic-host.example.com
      WebLogicPort 7499
   </Location>

</VirtualHost>
```

C

Enabling LDAP Synchronization in Oracle Identity Governance

This appendix explains how to manually configure LDAP synchronization of Oracle Identity Management with the LDAP identity store post-installation.
LDAP synchronization works when Oracle Identity Manager is integrated with Access Manager (OAM). But OAM-OIM integration using IDMConfigTool is not supported in 12c. The integration will be based on LDAP connectors and will be available post PS3. However, if you have upgraded from Release 11.1.2.3 to Release 12.2.1.3, then you can continue with LDAP synchronization as described in Enabling LDAP Synchronization in Oracle Identity Manager in the *Integration Guide for Oracle Identity Management Suite* for Release 11.1.2.3.

D

Configuring Oracle Virtual Directory for Integration with Oracle Access Management Access Manager

This appendix explains how to configure Oracle Virtual Directory for integration with Oracle Access Management Access Manager (Access Manager).

> **Note:**
>
> Using Oracle Virtual Directory with Access Manager is *optional*, so the procedures described here are not required as part of the core integration process.

This appendix includes the following sections:

- Creating and Configuring Oracle Virtual Directory Adapters
- Using the OAMPolicyControl Plug-In with Oracle Access Manager 10g

> **Note:**
>
> You can use Oracle Virtual Directory with most LDAP-enabled technologies. The information contained in this appendix highlights Oracle Virtual Directory features and capabilities that simplify common integrations.
>
> For assistance with other Oracle Virtual Directory integrations, contact your Oracle support representative.

D.1 Creating and Configuring Oracle Virtual Directory Adapters

To configure Oracle Virtual Directory for integration with Access Manager, you use the Oracle Directory Services Manager's Setup for Oracle Access Manager Quick Config Wizard. This Wizard walks you through the steps to create the required Local Store Adapter and the appropriate adapter type (LDAP, Database, or Custom) for the data repository used by Access Manager.

1. Log in to Oracle Directory Services Manager.
2. Select **Advanced** from the task selection bar. The Advanced navigation tree appears.
3. Expand the **Quick Config Wizards** entry in the Advanced tree.
4. Click **Setup for Oracle Access Manager** in the tree. The Setup for Oracle Access Manager screen appears.

5. Enter the namespace for the Local Store Adapter in DN format in the Namespace used for creating Local Store Adapter (LSA) field and click **Apply**. The Adapters screen appears.
6. Create an adapter that is appropriate for the data repository that Access Manager uses. Refer to the sections listed at the end for instructions:
7. Configure the adapter for the data repository that Access Manager uses by selecting **Adapter** from the Oracle Directory Services Manager task selection bar and then clicking the name of the adapter to configure in the Adapter tree.

> **See Also:**
>
> Refer to the following sections for information about configuring each type of adapter:
>
> - Creating and Configuring an LDAP Adapter
> - Creating and Configuring a Database Adapter
> - Configuring Custom Adapters

D.1.1 Creating and Configuring an LDAP Adapter

To create an LDAP Adapter for Access Manager, refer to "Creating LDAP Adapters" in the Oracle® Fusion Middleware Administrator's Guide for Oracle Virtual Directory.

After you create the LDAP Adapter, you can configure that adapter by using the procedures described in the following sections:

- Configuring LDAP Adapter General Settings
- Managing Certificate Authorities for LDAP Adapters Secured by SSL

> **Note:**
>
> For more information, about configuring LDAP adapters, refer to "Configuring LDAP Adapters" in the Oracle® Fusion Middleware Administrator's Guide for Oracle Virtual Directory.

D.1.1.1 Configuring LDAP Adapter General Settings

You can configure the general settings for the adapter by clicking the adapter name in the Adapter tree, clicking the **General** tab, setting values for the following fields, and clicking **Apply**:

Root
This field defines the root DN that the adapter provides information for. The DN defined, and the child entries below it, comprise the adapter's namespace. The value you enter in this field should be the base DN value for the returned entries. For example, if you enter dc=mydomain,dc=com in the field, all entries end with dc=mydomain,dc=com.

Active
You can configure an adapter as active (enabled) or inactive (disabled). An adapter configured as inactive does not start during a server restart or an attempted adapter start. Use the inactive setting to keep old configurations available or in stand-by without having to delete them from the configuration. The default setting is active (enabled).

LDAP Server Details

Perform the following procedures to configure the proxy LDAP host information in the LDAP Servers table in the General tab. Each proxy LDAP host must provide equivalent content, that is, must be replicas.

Be careful when specifying only a single host for proxying. Without a failover host, the LDAP Adapter cannot automatically fail over to another host. A single host is suitable when Oracle Virtual Directory is connected to a logical LDAP service by using a load balancing system.

> **Note:**
>
> The information in the LDAP Servers table is used only if you set the Use DNS for Auto Discovery parameter to **No**.

To add a proxy LDAP host to the adapter:

1. Click the **Add Host** button.
2. Enter the IP Address or DNS name of the LDAP host to proxy to in the Hosts field.

> **Note:**
>
> Oracle Virtual Directory 12*c* (12.2.2) supports IPv6. If your network supports IPv6 you can use a literal IPv6 address in the Hosts field to identify the proxied LDAP host.

3. Enter the port number the proxied LDAP host provides LDAP services on in the Port field.
4. Enter a number between 0 and 100 in the Percentage field to configure the load percentage to send to the host. If the combined percentages for all of the hosts configured for the adapter do not total 100, Oracle Virtual Directory automatically adjusts the load percentages by dividing the percentage you entered for a host by the total percentage of all hosts configured for the adapter. For example, if you have three hosts configured for the adapter at 20 percent, 30 percent, and 40 percent, Oracle Virtual Directory adjusts the 20 to 22 (20/90), the 30 to 33 (30/90), and the 40 to 44 (40/90).
5. Select the Read-only option to configure the LDAP Adapter to only perform search operations on the LDAP host. The LDAP Adapter automatically directs all modify traffic to read/write hosts in the list.

To delete a proxy LDAP host from the adapter:

1. Click anywhere in the row of the host you want to delete in the Remote Host table.
2. Click the **Delete** button. A confirmation dialog box appears.
3. Click Confirm to delete the proxy LDAP host from the adapter.

To validate a proxy LDAP host connection:

1. Click anywhere in the row of the Remote Host table for the host you want to validate the connection for.
2. Click the **Validate** button. The connection to the proxy LDAP host must be validated for the adapter to proxy the LDAP host.

Use SSL/TLS
Enabling this option secures the communication between the LDAP Adapter and the proxy LDAP hosts using SSL/TLS.

> **See:**
>
> "Managing Certificate Authorities for LDAP Adapters Secured by SSL" for information on Certificate Authorities.

SSL Authentication Mode
If you select (enable) the **Use SSL/TLS** option, choose the SSL authentication mode to use for securing the adapter by selecting an option from the SSL Authentication Mode list. The SSL Authentication Mode setting is functional only when the Use SSL/TLS option is enabled.

Failover Mode
If set to **Sequential**, the first host specified in LDAP Servers table is used unless a failure occurs. If a failure occurs, the next host is tried. Sequential failover is often used for fail-over between geographies. In sequential failover, the LDAP Adapter attempts to use the designated host until it fails. At this point, it would fail-over to an equivalent host available in another data center or continent.
If set to **Distributed**, each new connection made is load balanced through the list defined by the LDAP Servers table. Distributed failover is most often used when proxying a set of LDAP hosts that are typically in the same data center or are equally available in terms of network performance.

> **Note:**
>
> If a remote host's network fails, a delay of several minutes may occur in Oracle Virtual Directory because of platform specific TCP socket timeout settings. However, Oracle Virtual Directory failover is operating properly and no data is lost during the delay.

Extended Trying
Enable this option to force the Oracle Virtual Directory server to continue trying to connect to the last host listed in the LDAP Servers table for new incoming requests on the adapter even after it has been determined that the connection to the host failed. When enabled, the adapter's **Heartbeat Interval** setting is ignored regardless if a connection to the host has failed and the host will not be removed from the LDAP Servers table. Some environments with distributed directories may prefer to disable the **Extended Trying** option with the **Routing Critical** setting to quickly return partial results at that time. The default setting is enabled.

Heartbeat Interval
The LDAP Adapter periodically verifies the availability of each the hosts defined in the LDAP Servers table. Any currently disabled host can be resurrected or a currently active host that fails the TCP/IP connection test is labeled as **false** during this verification cycle. The Heartbeat Interval parameter specifies the number of seconds between verification passes. Setting a value too low can cause unnecessary connections to the remote directory. Setting a value too high can mean extended time for recovery detection when you have a failure. For production environments, Oracle suggests starting with a value of 60 seconds, then making adjustments as needed.

Operation Timeout
The amount of time in milliseconds the server waits for an LDAP request to be acknowledged by a remote host. If the operation fails, the LDAP Adapter automatically tries the next server in the Remote Host table. The minimum configurable value is 100. Settings that are too low can cause erroneous failures on busy servers. For production environments, Oracle suggests starting with a value of 5000, which is 5 seconds, then making adjustments as needed.

Max Pool Connections
A tuning parameter that enables you to control how many simultaneous connections can be made to a single server. For production environments, Oracle suggests starting with a value of 10 connections, then making adjustments as needed.

Max Pool Wait
The maximum amount a time in milliseconds that an LDAP operation waits to use an existing connection before causing the LDAP Adapter to generate a new connection. For production environments, Oracle suggests starting with a value of 1000, which is 1 second, then making adjustments as needed.

Max Pool Tries
Maximum number of times an operation waits for an LDAP connection before overriding the Max Pool Connections parameter to generate a new connection. Maximum time is a function of multiplying Max Pool Wait time by the number of tries. If pool wait is 1 second, and 10 is the maximum number of tries, then if after 10 seconds an LDAP connection is not available in the normal pool, the pool will be expanded to handle the extended load. To prevent pool expansion beyond Max Pool Connections, set the number of tries to a high number. For production environments, Oracle suggests starting with a value of 10, then making adjustments as needed.

Use Kerberos
If you enable the **Use Kerberos** option:
You must set the Pass Through option to **BindOnly** because the Kerberos authentication can only be used to validate credentials and not passed to the back-end server for any other operation.
The RDN value must be the same as the Kerberos principal name, for example, sAMAccountName in Active Directory. This may mean that the bind DN for a Kerberos bind is not the actual user DN. For example, if the user DN is `cn=Jane Doe,cn=users,dc=mycompany,dc=com` but the sAMAccountName is `jdoe`, the bind DN with the Use Kerberos option enabled is `cn=jdoe,cn=users,dc=mycompany,dc=com`.
You must create a krb5.conf file and place it in the Oracle Virtual Directory's configuration folder. The krb5.conf has the following properties:

Property	Description
default_realm	The default domain used if not supplied by the mapping. For example, if a user binds as `uid=jsmith,ou=people,dc=myorg,dc=com`, this will be treated as `jsmith@myorg.com`. If the mapped namespace does not include a domain component (dc) based root, this value is substituted instead.
domain_realm	Defines a mapping between a domain and a realm definition. For example: `.oracle.com = ORACLE.COM`
realms	Defines one or more realms, for example: `ORACLE.COM = {...}`
kdc	The DNS name of the server running the Kerberos service for a particular realm definition.

Kerberos binds use the Kerberos libraries provided in the standard Java package. The Kerberos libraries use the krb5.conf file, which is not currently synchronized with Oracle Virtual Directory LDAP Adapter settings. The default libraries control Kerberos fail-over. Refer to Java documentation for more information on fail-over and advanced krb5.conf file configurations.

> **Note:**
>
> If a Microsoft Active Directory server is in the process of shutting down (either stopping or rebooting) and Oracle Virtual Directory tries to connect to it, Active Directory may not validate the credential and may return a `Client not Found in Kerberos Database` error message instead of returning a Key Distribution Center (Domain Controller) connection error.
> The end-user should attempt to login again and assuming that either the Active Directory server is available or Key Distribution Center fail-over is enabled, successful authentication should be returned.

Kerberos Retry
If you enable the **Use Kerberos** option, you can use the **Kerberos Retry** option to control whether Oracle Virtual Directory should retry logging in after failed authentication attempts. If you enable the **Kerberos Retry** option and authentication fails, Oracle Virtual Directory reloads the kerb5.conf file and retries the log in.

> **Note:**
>
> If you identified multiple Active Directory servers in a single Kerberos realm in the krb5.conf file, do not enable the **Kerberos Retry** option, as enabling the retry may disrupt fail-over functionality.

Use DNS For Auto Discovery
Instead of configuring specific proxy LDAP hosts in the LDAP Servers table, you can use this option to instruct Oracle Virtual Directory to use DNS to locate the

appropriate LDAP servers for the remote base defined, also known as serverless bind mode. The LDAP Adapter supports the following modes of operation:

- **No**: Use the LDAP Servers table configuration—no serverless bind.
- **Standard**: Use standard DNS lookup for a non-Microsoft server. All servers are marked as read/write, so enabling the **Follow Referrals** setting is advised to allow for LDAP write support.
- **Microsoft**: The DNS server is a Microsoft dynamic DNS and also supports load-balancing configuration. If proxying to a Microsoft dynamic DNS server, this is preferred setting because of Oracle Virtual Directory's ability to auto-detect read/write servers compared to read-only servers.

> **Note:**
>
> Remote base should have a domain component style name when using this setting, for example, dc=myorg,dc=com. This name enables Oracle Virtual Directory to locate the LDAP hosts within the DNS service by looking up myorg.com.

The following fields appear in the Settings section of the General tab:

Remote Base
The location in the remote server directory tree structure to which the local Oracle Virtual Directory root suffix corresponds. This is the location in the remote directory under which Oracle Virtual Directory executes all searches and operations for the current adapter. The LDAP Adapter applies an automatic mapping of all entries from the remote base to the adapter root base.

DN Attributes
List of attributes to be treated as DNs for which namespace translation is required, such as member, uniquemember, manager. For example, when reading a group entry from a proxied directory, Oracle Virtual Directory automatically converts the DN for the group entry itself and the uniquemember or member attributes if these attributes are in the DN Attributes list.

> **Note:**
>
> Translate only those attributes you know must be used by the client application. Entering all possible DN attributes may not be necessary and can consume some a small amount of additional CPU time in the proxy.

To add attributes to the DN Attributes list:

1. Click **Add**. The Select DN Attribute dialog box appears.
2. Select the attribute you want to add.
3. Click **OK**.

Escape Slashes
When a / character is encountered in a directory, Oracle Virtual Directory can optionally escape the slashes with back-slashes \ character. Some directory server products accept un-

escaped slashes, while others reject them. Selecting this setting enables escaping of slashes.

Follow Referrals
Enabling this setting causes the LDAP Adapter to follow (chase) referrals received from a source directory on the client's behalf. If disabled, the referral is blocked and not returned to the client.
The following list summarizes the LDAP Adapter's behavior with different settings in relation to the send managed DSA control in LDAP operations setting:

- If the LDAP Adapter's Follow Referrals is set to **Enabled (true)**, and Send Managed DSA Control in LDAP Operations is also set to **True**, Oracle Virtual Directory does not chase the referral entries, but it returns them back to the client.
- If the LDAP Adapter's Follow Referrals is set to **Enabled (true)**, but Send Managed DSA Control in LDAP Operations is set to **False**, Oracle Virtual Directory chases the referral entries.
- If the LDAP Adapter's Follow Referrals is set to **Disabled (false)**, but Send Managed DSA Control in LDAP Operations is set to **True**, Oracle Virtual Directory does not chase the referral entries, but it returns them back to the client.
- If the LDAP Adapter's Follow Referrals is set to **Disabled (false)**, and Send Managed DSA Control in LDAP Operations is also set to **False**, Oracle Virtual Directory does not chase the referral entries and does not return them back to client.

Proxied Page Size
If enabled, this setting allows the proxy to use the paged results control with a proxied directory. Enabling this setting is most often used when a directory limits the number of results in a query. This setting is used on behalf of and transparently to Oracle Virtual Directory's clients.

The following fields appear in the Credential Processing section of the General tab:

Proxy DN
The default DN that the LDAP Adapter binds with when accessing the proxied directory. Depending on the **Pass-through Mode** setting, this DN is used for all operations, or only for exceptional cases such as pass-through mode. The form of the distinguished name should be in the form of the remote directory. Empty values are treated as Anonymous.

Proxy Password
The authentication password to be used with the **Proxy DN** value. To set the password, enter a value in clear text. When loaded on the server, the value is automatically hashed with a reversible mask to provide additional security, for example, {OMASK}jN63CfzDP8XrnmauvsWs1g==.

Pass-through Mode
To pass user credentials presented to Oracle Virtual Directory to the proxied LDAP server for all operations, set to **Always**. To pass user credentials to the proxied LDAP server for bind only and use the default server credentials for all other operations, set to **Bind Only**. To use the Proxy DN credentials for all operations, set to **Never**.

> **Note:**
>
> In some situations when pass-through mode is set to **Always**, the LDAP Adapter may still use the Proxy DN. This occurs when the user credential cannot be mapped, for example, from another adapter namespace, or is the root account. If defining multiple adapters to different domain controllers within a Microsoft Active Directory forest, you can program the LDAP Adapter to proxy credentials from other adapters (that is, two or more adapters pointing to the same Active Directory forest) by using the **Routing Bind-Include** setting.

The following fields appear in the Ping Protocol Settings section of the General tab:

The Ping Protocol Settings provide options for how to determine when a source LDAP directory server that is not responding becomes available. If multiple source directory servers are configured, Oracle Virtual Directory identifies the non-responsive servers and performs subsequent operations against the next available server.

Ping Protocol
Select either **TCP** or **LDAP** as the protocol Oracle Virtual Directory should use to ping source directory servers. Select **LDAP** if the source directory server is using SSL.

> **Note:**
>
> While the **TCP** protocol option is faster than the **LDAP** option, it may produce an inaccurate response from the source directory server if its network socket is available, but its LDAP server process is unavailable.

Ping Bind DN
If you select **LDAP** as the Ping Protocol, identify the DN to use for the LDAP bind.

Ping Bind Password
If you select **LDAP** as the Ping Protocol, identify the password for the DN specified in the Ping Bind DN setting.

D.1.1.2 Managing Certificate Authorities for LDAP Adapters Secured by SSL

In some situations, SSL connections from Oracle Virtual Directory to the SSL port of an LDAP Adapter can fail and the following message may appear:

```
Oracle Virtual Directory could not load certificate chain
```

Two examples of situations when this may happen are when:

- you create a new LDAP Adapter secured by SSL and use an untrusted Certificate Authority
- a certificate for an existing LDAP Adapter secured by SSL expires and the new certificate is signed by an untrusted Certificate Authority

To resolve this issue, import the LDAP server certificate *and* the Root Certificate Authority certificate used to sign the LDAP server certificate, into the Oracle Virtual Directory server so it knows the certificates are trusted.

Use the following `keytool` command and an appropriate alias **all on one command line**:

```
ORACLE_HOME/jdk/jre/bin/keytool -import -trustcacerts
-alias "NEW_CA" -file PATH_TO_CA_CERTIFICATE
-keystore ORACLE_INSTANCE/config/OVD/ovd1/keystores/adapters.jks
```

Using LDAP Adapters with Microsoft Active Directory and Microsoft Certificate Services

By default, Microsoft Certificate Services automatically update expired Active Directory SSL certificates. However, client applications are not normally notified of this change. If this happens, the Oracle Virtual Directory LDAP Adapter connected to an updated Active Directory server stops functioning. If this occurs, use Oracle Directory Services Manager to configure the LDAP Adapter to import trusted certificates and the adapter should begin to function again.

> **Note:**
>
> Active Directory servers only support SSL server authentication. For this reason, you are only required to load the root CA certificate of the Certification Authority that signed the Active Directory server certificate to the OVD keystore. If the Active Directory server certificate is also loaded, then based on the standard behavior of Sun JSSE, OVD does not execute an expiry check of the trusted certificate.
>
> Consequently, if the certificate sent by the back-end LDAP server is stored as a trusted certificate in the OVD keystore, no expiry check is executed.

D.1.2 Creating and Configuring a Database Adapter

To create a Database Adapter for Access Manager, refer to "Creating Database Adapters" in the Oracle® Fusion Middleware Administrator's Guide for Oracle Virtual Directory.

After you create the Database Adapter, you can configure the general settings for that adapter by clicking the adapter name in the Adapter tree, clicking the **General** tab, setting values for the following fields, and clicking **Apply**:

> **Note:**
>
> For more information, about configuring LDAP adapters, refer to "Configuring Database Adapters" in the Oracle® Fusion Middleware Administrator's Guide for Oracle Virtual Directory.

Root
This field defines the root DN that the adapter provides information for. The DN defined, and the child entries below it, comprise the adapter's namespace. The value you enter in this field should be the base DN value for returned entries. For example, if you enter dc=mydomain,dc=com in the field, all entries end with dc=mydomain,dc=com.

Active
An adapter can be configured as active (enabled) or inactive (disabled). An adapter configured as inactive does not start during a server restart or an attempted adapter start. Use the inactive setting to keep old configurations available or in stand-by without having to delete them from the configuration. The default setting is active.

The following fields appear in the Connection Settings section of the General tab:

URL Type
Select an option from the following URL Type list. Some fields for Database Adapter connection settings differ depending on which option you choose. After selecting an option, continue configuring the Connection Settings by setting the fields listed for each option.

- **Use Custom URL**: Select this option to connect Oracle Virtual Directory a custom database.
  - Enter the JDBC driver class name for the database in the JDBC Driver Class field.
  - Enter the URL that Oracle Virtual Directory should use to access the database in the Database URL field.
  - Enter the user name that the Database Adapter should use to connect the database in the Database User field.
  - Enter the password for the user name you entered in the Database User field in the Password field. Oracle Virtual Directory replaces the value you enter in this field with a reversible masked value upon startup.
- **Use Predefined Database**: Select this option to connect to a predefined database. The predefined databases appear in the Database Type list after selecting Use Predefined Database from the URL Type list. If you are unsure if Oracle Virtual Directory has predefined your type of database, select Use Predefined Database from the URL Type list and verify if your database is listed in the Database Type list. If your database is listed in the Database Type list, continue with the following steps. If your database is not listed, select **Use Custom URL** from the URL Type list and perform the steps for using a custom URL.
  - Select the type of your database from the Database Type list. After selecting the database type, the JDBC Driver Class and Database URL fields are populated with the appropriate information for the database.
  - Enter the IP Address or DNS host name of the database in the Host field.
  - Enter the port number the database listens on in the Port field.
  - Enter the name of the database, for example, the Oracle SID, in the Database Name field.
  - Enter the user name that the Database Adapter should use to connect the database in the Database User field.
  - Enter the password for the user name you entered in the Database User field in the Password field. Oracle Virtual Directory replaces the value you enter in this field with a reversible masked value upon startup.

The following fields appear in the Settings section of the General tab:

Ignore Modify Objectclass
Since objectclasses in the database are logical objects and do not map directly to a table column in the mapping, modifications to the objectclass attribute can cause errors. If the **Ignore Modify Objectclasses** option is enabled, the Database Adapter removes any

references to the objectclass attribute so that errors are *not* be sent to the client application, that is, they are ignored. If the **Ignore Modify Objectclasses** option is not selected, error messages *are* sent to the client application

Include Object Class Super Classes
This setting causes the Database Adapter to list objectclass parent classes along with the main objectclass in the objectclass attribute. Disable this setting when you want to emulate Microsoft Active Directory server schema. For most scenarios, it is useful to enable this setting so that objectclass=xxx queries can be executed against parent objectclass values.

Enable Case Insensitive Search
Enabling (selecting) the **Enable Case Insensitive Search** option makes the search case insensitive for case insensitive LDAP attributes, such as uid. Oracle Virtual Directory uses UPPER in the SQL query when **Enable Case Insensitive Search** is enabled. If the database cannot maintain functional indexes, such as for Oracle TimesTen or MySQL databases, then you should disable the **Enable Case Insensitive Search** option. When the **Enable Case Insensitive Search** is disabled, Oracle Virtual Directory performs case sensitive searches and does not use UPPER in the SQL query. The default value for **Enable Case Insensitive Search** is Enable.

Maximum Connections
This setting defines the maximum connections the Database Adapter may make with the database.

Connection Wait Timeout
This setting determines how much time (in seconds) the Database Adapter should wait before timing-out when trying to establish a connection with the database.

The following fields appear in the DB/LDAP Mapping section of the General tab:

Used Database Tables
This field displays the database tables the Database Adapter is set to use. To add a database table, click the **Add** button, navigate to the table file, select it and click **OK**.

The following fields appear in the Object Classes section of the General tab:

Object Classes
This field displays object classes and their RDNs that map to the database tables. To add an Object Class Mapping, click the **Create** button, select the appropriate object class from the Object Class list, enter an RDN value for the object class in the RDN field, and click **OK**.

> **Note:**
>
> For more information, about configuring Database adapters, refer to "Configuring Database Adapters" in the Oracle® Fusion Middleware Administrator's Guide for Oracle Virtual Directory.

D.1.3 Configuring Custom Adapters

To create a Custom Adapter for Access Manager, refer to "Creating Custom Adapters" in the Oracle® Fusion Middleware Administrator's Guide for Oracle Virtual Directory.

After you create the Custom Adapter you can configure the general settings for that adapter by clicking the adapter name in the Adapter tree, clicking the **General** tab, setting values for the following fields, and clicking **Apply**:

> **Note:**
>
> For more information, about configuring LDAP adapters, refer to "Configuring Custom Adapters" in the Oracle® Fusion Middleware Administrator's Guide for Oracle Virtual Directory.

Root
This field defines the root DN that the adapter provides information for. The DN defined, and the child entries below it, comprise the adapter's namespace. The value you enter in this field should be the base DN value for returned entries. For example, if you enter dc=mydomain,dc=com in the field, all entries end with dc=mydomain,dc=com.

Active
An adapter can be configured as active (enabled) or inactive (disabled). An adapter configured as inactive does not start during a server restart or an attempted adapter start. Use the inactive setting to keep old configurations available or in stand-by without having to delete them from the configuration. The default setting is active.

D.2 Using the OAMPolicyControl Plug-In with Oracle Access Manager 10g

> **Note:**
>
> This section is only relevant to customers that are still running Oracle Access Manager **10g**. The OAMPolicyControl plug-in does not work with Access Manager 11g.

Oracle Virtual Directory provides the OAMPolicyControl plug-in to simplify the Oracle Virtual Directory-Access Manager 10g integration for applications that use LDAP for authentication and want to use Access Manager policy controls, but cannot integrate with Access Manager.

D.2.1 Preparing to Deploy the OAMPolicyControl Plug-in

Before deploying the OAMPolicyControl plug-in, you must:

- Set the Bind pass-through settings to Never for any LDAP Adapters that are using the Access Manager policy configuration.

  The plug-in handles all authentications and uses proxy credentials to perform all operations.
- Configure different adapters for Access Manager.

These adapters should use the OAMPolicyControl plug-in to use Access Manager policies. If you deploy these adapters on the same Oracle Virtual Directory server, you must configure one of the following options:

  - Use a different LDAP namespace for each adapter. An Access Manager adapter namespace must be independent from the namespaces used by general purpose LDAP clients.
  - Use an Oracle Virtual Directory view, with accessibility criteria that distinguishes requests for different Access Manager adapters.

- Configure the Access Manager Access Server by:
  - Creating a proxy resource that corresponds to Oracle Virtual Directory.
  - Disabling the policy domains for Identity Server and Access Server because the plug-in does not cache the OBSSO Cookie.
- Configure the AccessSDK as follows:
  - Configure an AccessSDK installation for the Access Manager Access Server by using AccessServerSDK\oblix\tools\configureAccessGate.
  - Configure the opmn to start the Oracle Virtual Directory component by pointing the -Djava.library.path to the AccessSDK installation.

    Edit the INSTANCE_HOME/config/OPMN/opmn/opmn.xml file as follows:

```
<ias-component id="ovd1">
 <process-type id="OVD" module-id="OVD">
  <module-data>
   <category id="start-options">
    <data id="java-bin" value="$ORACLE_HOME/jdk/bin/java"/>
    <data id="java-options" value="-server -Xms512m -Xmx512m
     -Dvde.soTimeoutBackend=0
     -Doracle.security.jps.config=$ORACLE_
INSTANCE/config/JPS/jps-config-jse.xml
     -Djava.library.path=AccessSDK_install_
dir/AccessSDK/AccessServerSDK/oblix/lib/"/>
     <data id="java-classpath" value="$ORACLE_
HOME/ovd/jlib/vde.jar$:$ORACLE_HOME/jdbc/lib/ojdbc6.jar"/>
    </category>
   </module-data>
  <stop timeout="120"/>
 </process-type>
</ias-component>
```

  - Copy the jobaccess.jar file from AccessSDK_install_dir/AccessServerSDK/oblix/lib to ORACLE_HOME/ovd/plugins/lib.

> **Note:**
>
> Failure to successfully complete the preceding prerequisite configurations will cause the Oracle Virtual Director to generate a NoClassDefFound error.

D.2.2 Configuration Parameters of the OAMPolicyControl Plug-in

The OAMPolicyControl plug-in has the following configuration parameters:

> **Note:**
>
> All of the following configuration parameters—except for useAccessAuthPolicy—are required to deploy the OAMPolicyControl plug-in.

resourceIdOVD
Identifies the proxy resource for Oracle Virtual Directory that the Access Manager server configures. For example: //*host*:*port*/ovd_proxy_resource.

identityproxyid
Used for authentication against the Identity Server, the identityproxyid parameter identifies the value of the administrator's usernameAttribute.

install_dir
Identifies the AccessSDK installation directory containing the required libraries. For example: *AccessSDK_INSTALL_DIRECTORY*/AccessServerSDK/.

OrclOVDEncryptedproxypasswd
Administrator password for authentication against Identity Server.

identityEndpointAddress
Identifies the URL corresponding to the listening endpoint of the Identity Server's um_modifyUser web service. For example: http://*host*:*port*/identity/oblix/apps/userservcenter/bin/userservcenter.cgi

usernameAttribute
Identifies the attribute configured to be the Login attribute of the Identity Server. For example, uid or genUserId.

useAccessAuthPolicy
An optional and case-insensitive parameter, useAccessAuthPolicy determines usage of the Access Manager server's authorization policies while accessing the proxy resource. Supported values are True and False. The default setting is False.