Oracle® Identity Management

Integration Guide

10*g* Release 2 (10.1.2)

B14085-02

July 2005

ORACLE®

Oracle Identity Management Integration Guide, 10*g* Release 2 (10.1.2)

B14085-02

Primary Author: Don Gosselin

Contributor: Vasuki Ashok, Tridip Bhattacharya, Neelima Bawa, Ramakrishna Bollu, Margaret Chou, Saheli Dey, Rajinder Gupta, Ajay Keni, Ashish Kolli, Stephen Lee, David Lin, Michael Mesaros, Radhika Moolky, Richard Smith, Hari Sastry, David Saslav, Ramaprakash Sathyanarayan, Bhupindra Singh, Gurudatt Shakshikumar, Amit Sharma, Jason Sharma, Daniel Shih, Saurabh Shrivastava, Uppili Srinivasan, Olaf Stullich, Dipankar Thakuria, Sivakumar Venugopal

Contents

Part II General Administration of Oracle Directory Integration and Provisioning

3 Oracle Directory Integration and Provisioning Administration Tools

4 Managing the Oracle Directory Integration and Provisioning Server

7 Administration of Directory Synchronization

8 Bootstrapping of a Directory in Oracle Directory Integration and Provisioning

9 Synchronization with Relational Database Tables

10 Synchronization with Oracle Human Resources

11 Synchronization with Third-Party Metadirectory Solutions

Part IV Provisioning in Oracle Identity Management

12 Oracle Provisioning Service Concepts

18 Integration with the Microsoft Active Directory Environment

19 Integration with the Microsoft Windows NT 4.0 Environment

20 Integration with SunONE (iPlanet) Directory Server

List of Figures

List of Tables

Preface

Oracle Identity Management Integration Guide describes the features, architecture, and administration of Oracle Internet Directory. For information about installation, see the installation documentation for your operating system.

Audience

Oracle Identity Management Integration Guide is intended for anyone who performs administration tasks for the Oracle Internet Directory. You should be familiar with either the UNIX operating system or the Microsoft Windows NT operating system in order to understand the line-mode commands and examples. You can perform all of the tasks through the line-mode commands, and you can perform most of the tasks through Oracle Directory Manager, which is operating system-independent.

To use this document, you need some familiarity with the **Lightweight Directory Access Protocol (LDAP)**.

Documentation Accessibility

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

`http://www.oracle.com/accessibility/`

Accessibility of Code Examples in Documentation

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

Accessibility of Links to External Web Sites in Documentation

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

TTY Access to Oracle Support Services

Oracle provides dedicated Text Telephone (TTY) access to Oracle Support Services within the United States of America 24 hours a day, seven days a week. For TTY support, call 800.446.2398.

Related Documentation

For more information, see:

- Online help available through Oracle Directory Manager, the Oracle Delegated Administration Services and Oracle Enterprise Manager 10*g*
- The Oracle Application Server and Oracle Database documentation sets, especially:
  - *Oracle Identity Management Application Developer's Guide*
  - *Oracle Identity Management Concepts and Deployment Planning Guide*
  - *Oracle9i Database Administrator's Guide*
  - *Oracle9i Application Developer's Guide - Fundamentals*
  - *Oracle Application Server Administrator's Guide*
  - *Oracle Identity Management Guide to Delegated Administration*
  - *Oracle9i Net Services Administrator's Guide*
  - *Oracle9i Real Application Clusters Administration*
  - *Oracle9i Advanced Replication*
  - *Oracle Advanced Security Administrator's Guide*
  - *Oracle Internet Directory Administrator's Guide*
  - *Oracle Identity Management User Reference*
  - *Oracle Application Server Single Sign-On Administrator's Guide*
  - *Oracle Application Server Certificate Authority Administrator's Guide*

For additional information, see:

- Chadwick, David. *Understanding X.500—The Directory.* Thomson Computer Press, 1996.
- Howes, Tim and Mark Smith. *LDAP: Programming Directory-enabled Applications with Lightweight Directory Access Protocol.* Macmillan Technical Publishing, 1997.
- Howes, Tim, Mark Smith and Gordon Good, *Understanding and Deploying LDAP Directory Services.* Macmillan Technical Publishing, 1999.
- Internet Assigned Numbers Authority home page, `http://www.iana.org` for information about object identifiers
- Internet Engineering Task Force (IETF) documentation available at: `http://www.ietf.org`, especially:
  - The LDAPEXT charter and LDAP drafts
  - The LDUP charter and drafts
  - RFC 2254, "The String Representation of LDAP Search Filters"
  - RFC 1823, "The LDAP Application Program Interface"

- The OpenLDAP Community, http://www.openldap.org

Conventions

The following text conventions are used in this document:

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

What's New in Oracle Identity Management Integration?

This section provides a brief description of new features introduced with the latest releases of Oracle Internet Directory, and points you to more information about each one. It contains these topics:

New Features Introduced with Oracle Application Server 10*g* Release 2 (10.1.2)

This section describes the new features introduced with Oracle Application Server 10*g* Release 2 (10.1.2).

- Enhanced provisioning capabilities and functionality—This release includes enhanced capabilities and functionality with the Oracle Provisioning Service. You can also use the new Oracle Internet Directory Provisioning Console, a graphical interface for administrators to provision users in Oracle Internet Directory. The Provisioning Console was created with Oracle Delegated Administration Services, and works alongside the Oracle Internet Directory Self-Service Console. For more information, see Part IV, "Provisioning in Oracle Identity Management".
- **Graphical administration of the Oracle directory integration and provisioning server**—You can now use the new Oracle Directory Integration and Provisioning Server Administration, a Java-based utility for graphically administering the Oracle directory integration and provisioning server. For more information, see Chapter 3, "Oracle Directory Integration and Provisioning Administration Tools".
- **Express configuration of the Microsoft Active Directory Connector**—You can now perform an express configuration of the Microsoft Active Directory connector. Express configuration uses default settings to automatically perform all required configurations, and also creates two synchronization profiles, one for import and one for export.

See Also:

Chapter 3, "Oracle Directory Integration and Provisioning Administration Tools"

Chapter 18, "Integration with the Microsoft Active Directory Environment"

- **Simplified configuration of Windows Native Authentication**—This guide now includes detail instructions for configuring Windows native authentication. For more information, see Chapter 18, "Integration with the Microsoft Active Directory Environment".

New Features Introduced with Oracle Internet Directory 10*g* (9.0.4)

This section describes the new features introduced with Oracle Internet Directory Release 10*g* (9.0.4).

- **Integration with the Microsoft Windows environment**—You can integrate the Oracle Application Server infrastructure with the Microsoft Windows Operating System—including Microsoft Active Directory and Microsoft Windows NT 4.0. This integration is achieved by using the Active Directory Connector in Oracle Directory Integration and Provisioning and plug-ins.

  See Also: Chapter 18, "Integration with the Microsoft Active Directory Environment"

- **External authentication support**—You can store user security credentials in a repository other than Oracle Internet Directory—for example, a database or another LDAP directory such as Microsoft Active Directory or SunONE Directory Server. You can then use these credentials for user authentication.

  See Also:

  - The chapter on setting up the customized external authentication password in the *Oracle Internet Directory Administrator's Guide*
  - "Choose Where to Store Passwords" on page 17-5

New Features Introduced with Oracle Internet Directory Release 9.0.2

This section describes the new features introduced with Oracle Internet Directory Release 9.0.2.

- **New directory integration capabilities**—Oracle Internet Directory Release 9.0.2 introduces new kinds of connectivity with other applications and repositories, both Oracle-built and otherwise. The new Oracle Provisioning Service and Oracle Directory Synchronization Service are built upon Oracle Directory Integration and Provisioning (introduced with Oracle Internet Directory v2.1.1.1 in the Oracle8*i* Release 3 time frame).
  - **Oracle Provisioning Service**—Provisioning is the process of granting or revoking a user's access to application resources based on business rules. The user may be either a human end user or an application.

    The Oracle Provisioning Service ensures that subscribing applications or business entities are alerted to updates in Oracle Internet Directory for keeping local repositories synchronized. It enables you to synchronize local,

application-specific information by using Oracle Internet Directory as a source of truth.

- **Oracle Directory Synchronization Service and the LDAP connector**—The Oracle Directory Synchronization Service enables near-complete leveraging of previously-deployed infrastructure, including but not limited to ERP and CRM systems, third-party LDAP directories, and NOS user repositories. It enables you to synchronize information between enterprise directories and Oracle Internet Directory. This allows for centralized administration, thereby reducing administrative costs. It ensures that data is consistent and up-to-date across the enterprise.

  Tip: Chapter 1, "Introduction to Oracle Identity Management Integration"

New Features Introduced with Oracle Internet Directory Release 3.0.1

This section describes the new features introduced with Oracle Internet Directory Release 3.0.1.

- **Oracle Directory Integration and Provisioning**—This new feature enables you to synchronize various directories with Oracle Internet Directory. It also makes it easier for third party metadirectory vendors and developers to develop and deploy their own connectivity agents.

New Features Introduced with Oracle Internet Directory Release 2.1.1

This section describes the new features introduced with Oracle Internet Directory release 2.1.1.

- **Synchronization with multiple directories in a metadirectory environment (release 2.1.1 only)**—If you are working in a metadirectory environment, then this new feature enables you to synchronize multiple directories with Oracle Internet Directory.

Note: This feature was replaced in Release 3.0.1 by Oracle Directory Integration and Provisioning. See Chapter 1, "Introduction to Oracle Identity Management Integration" for further information.

Part I

Getting Started with Oracle Identity Management Integration

This part discusses the concepts, components, architecture, and security features involved in Oracle Identity Management Integration. It contains these chapters

1

Introduction to Oracle Identity Management Integration

This chapter introduces Oracle Identity Management integration, its components, structure, and administration tools.

This chapter contains these topics:

- Why Oracle Identity Management Integration?
- Installation Options
- Synchronization, Provisioning, and the Difference Between Them
- Components Involved in Oracle Identity Management Integration

> **See Also:** Appendix B, "Case Study: A Deployment of Oracle Directory Integration and Provisioning" for an example of how you can deploy Oracle Identity Management integration

Why Oracle Identity Management Integration?

Oracle Identity Management enables you to reduce administrative time and costs by integrating your applications and directories—including third-party LDAP directories—with Oracle Internet Directory. It does this by using Oracle Directory Integration and Provisioning. For example, you might need to do the following:

- Keep employee records in Oracle Human Resources consistent with those in Oracle Internet Directory. Directory Integration and Provisioning provides this synchronization through the Oracle Directory Synchronization Service.
- Notify certain LDAP-enabled applications—such as OracleAS Portal—whenever changes are applied to Oracle Internet Directory. The Directory Integration and Provisioning provides this notification through Oracle Provisioning Service.

Throughout the integration process, Oracle Directory Integration and Provisioning ensures that the applications and other directories receive and provide the necessary information in a reliable way.

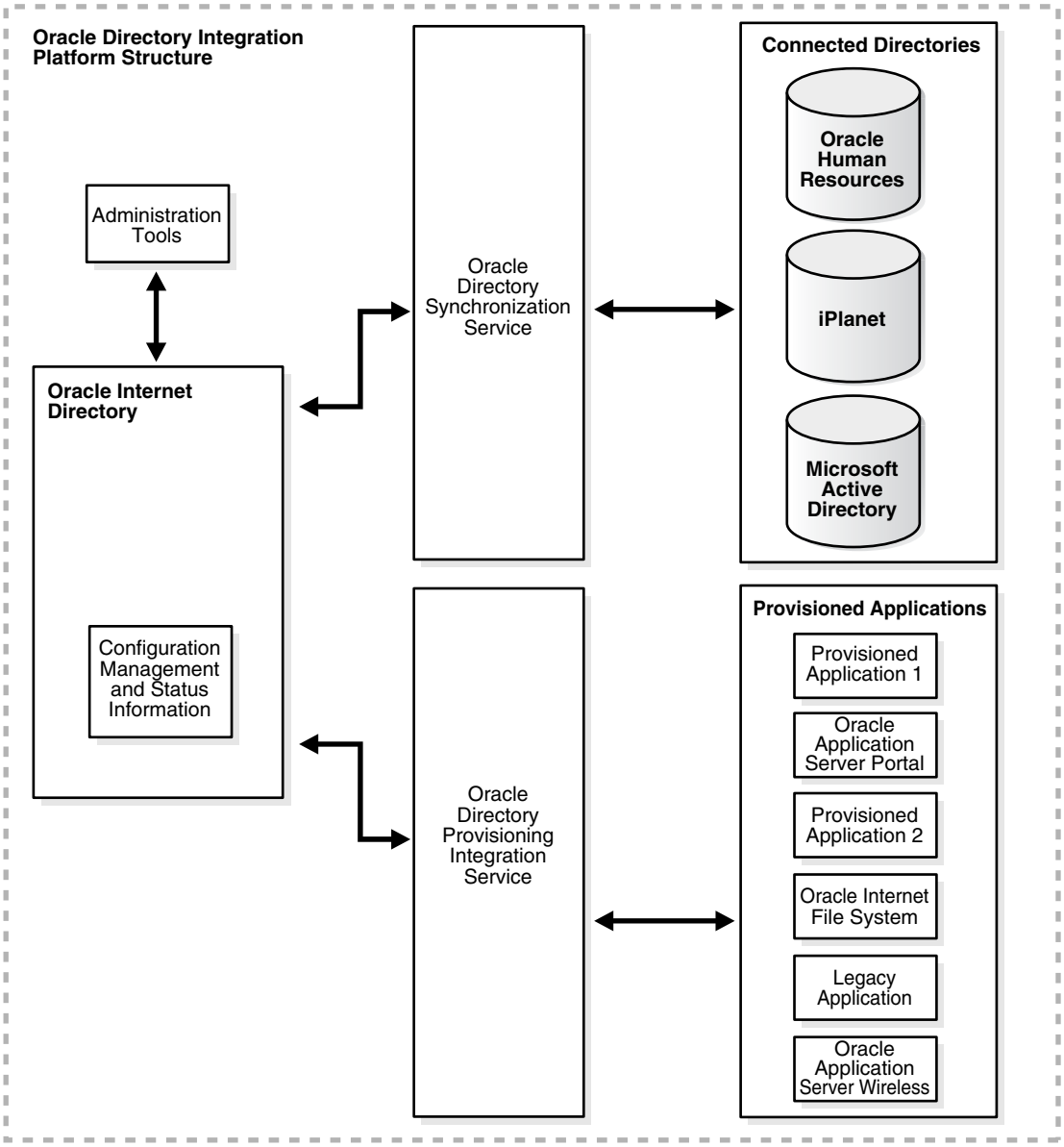
You can integrate with various directories, including Microsoft Active Directory and SunONE Directory Server. For example, in an Oracle Application Server environment, where access to Oracle components relies on data stored in Oracle Internet Directory, you can still use Microsoft Active Directory as the central enterprise directory. Users of that directory can still access Oracle components because Directory Integration and Provisioning can synchronize the data in Microsoft Active Directory with that in Oracle Internet Directory.

See Also:

- Chapter 10, "Synchronization with Oracle Human Resources"
- Chapter 18, "Integration with the Microsoft Active Directory Environment"
- Chapter 19, "Integration with the Microsoft Windows NT 4.0 Environment"
- Chapter 20, "Integration with SunONE (iPlanet) Directory Server"

Figure 1–1 on page 1-2 shows a sample deployment of Directory Integration and Provisioning.

Figure 1–1 Example of an Oracle Directory Integration and Provisioning Environment



In the example in Figure 1–1, Oracle Internet Directory is synchronized with connected directories by way of the Oracle Directory Synchronization Service. In this example, the connected directories are Oracle Human Resources, SunONE Directory Server, and Microsoft Active Directory. Similarly, changes in Oracle Internet Directory are sent to various applications by using the Oracle Provisioning Service. In this example, the provisioned applications include OracleAS Portal, Oracle Files, Oracle Application Server Wireless, two unspecified provisioned application, and a legacy application.

Installation Options

By default, Oracle Directory Integration and Provisioning is installed as a component of Oracle Internet Directory. However, you can also install Oracle Directory Integration and Provisioning in a standalone installation. You should install a standalone instance of Oracle Directory Integration and Provisioning under the following circumstances:

- When you need Oracle Internet Directory to run on a separate host for performance reasons
- When the applications that you need to provision and synchronize required intensive processing
- You need to run multiple instances of Oracle Directory Integration and Provisioning for high-availability

Synchronization, Provisioning, and the Difference Between Them

Synchronization has to do with directories rather than applications. It ensures the consistency of entries and attributes that reside in both Oracle Internet Directory and other connected directories.

Provisioning has to do with applications. It notifies them of changes to user or group entries or attributes that the application needs to track.

This section contains these topics:

Synchronization

Synchronization enables you to coordinate changes among Oracle Internet Directory and connected directories. For all directories to both use and provide only the latest data, each directory must be informed of change made in the other connected directories. Synchronization ensures that any change to directory information—including, but not limited to data updated through provisioning—is kept consistent.

Whenever you decide to connect a third-party directory to Oracle Internet Directory, you create a synchronization profile for that specific directory. This profile specifies the format and content of the data to be synchronized between Oracle Internet Directory and the connected directory. To create a synchronization profile, you use the Directory Integration and Provisioning Assistant.

See Also:

- Part III, "Synchronization in Oracle Identity Management Integration"
- The chapter on Oracle Directory Integration and Provisioning tools in the *Oracle Identity Management User Reference* for information on the Directory Integration and Provisioning Assistant

Provisioning

Provisioning enables you to ensure that an application is notified of directory changes to, for example, user or group information. Such changes can affect whether the application allows a user access to its processes and which resources can be used.

Use provisioning when you are designing or installing an application that

- Does not maintain a directory
- Is LDAP-enabled
- Can and should allow only authorized users to access its resources

When you install an application that you want to provision, you must create a provisioning integration profile for it by using the Provisioning Subscription Tool.

See Also:

- Part IV, "Provisioning in Oracle Identity Management"
- The chapter on Oracle Directory Integration and Provisioning tools in the *Oracle Identity Management User Reference* for information on the Provisioning Subscription Tool

How Synchronization and Provisioning Differ

Synchronization and provisioning have important operational differences as described in Table 1–1.

Table 1–1 Directory Synchronization and Provisioning Integration Distinctions

	Directory Synchronization	Provisioning Integration
The time for action	Application deployment time. Directory synchronization is for connected directories requiring synchronization with Oracle Internet Directory.	Application design time. Provisioning integration is for application designers developing LDAP-enabled applications.
Communication direction	Either one-way or two-way—that is, either from Oracle Internet Directory to connected directories, the reverse, or both	Two way—that is, from Oracle Internet Directory to provisioned applications, and from provisioned applications to Oracle Internet Directory
Type of data	Any data in a directory	Restricted to provisioned users and groups
Examples	Oracle Human Resources SunONE Directory Server Microsoft Active Directory	OracleAS Portal

Components Involved in Oracle Identity Management Integration

This section describes the components involved in Oracle Identity Management integration. It contains these topics:

- Oracle Internet Directory
- Oracle Directory Integration and Provisioning Server
- Oracle Application Server Single Sign-On

Oracle Internet Directory

Oracle Internet Directory is the repository in which Oracle components and third-party applications store and access user identities and credentials. It uses the Oracle directory server to authenticate users by comparing the credentials entered by users with the credentials stored in Oracle Internet Directory. When credentials are stored in a third-party directory and not in Oracle Internet Directory, users can still be authenticated. In this case, Oracle Internet Directory uses an external authentication plug-in that authenticates users against the third-party directory server.

Oracle Directory Integration and Provisioning Server

The Oracle directory integration and provisioning server is the shared server process that provides functionality for the Oracle Directory Synchronization Service and the Oracle Provisioning Service.

What the Oracle Directory Integration and Provisioning Server Does

The directory integration and provisioning server performs these services:

- Oracle Directory Synchronization Service:
  - Scheduling—Processing a synchronization profile based on a predefined schedule
  - Mapping—Executing rules for converting data between connected directories and Oracle Internet Directory
  - Data propagation—Exchanging data with connected directories by using a connector
  - Error handling
- Oracle Provisioning Service:
  - Scheduling—Processing a provisioning profile based on a predefined schedule
  - Event Notification—Notifying an application of a relevant change to the user or group data stored in Oracle Internet Directory
  - Error handling

Tip: Chapter 4, "Managing the Oracle Directory Integration and Provisioning Server"

About the Oracle Directory Synchronization Service

In the Oracle Directory Integration and Provisioning environment, the contents of connected directories are synchronized with Oracle Internet Directory through the Oracle Directory Synchronization Service.

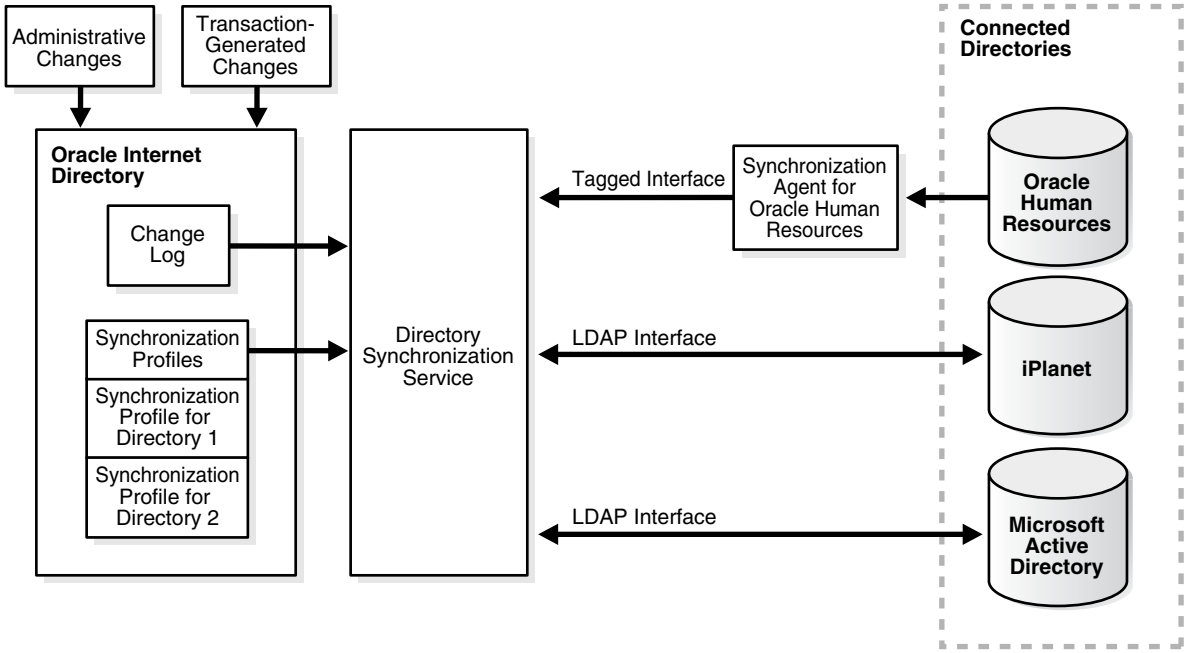
For Oracle Application Server components, Oracle Internet Directory is the central directory for all information, and all other directories are synchronized with it. This synchronization can be:

- One-way: Some connected directories only supply changes to Oracle Internet Directory and do not receive changes from it. This is the case, for example, with Oracle Human Resources, the primary repository and "source of truth" for employee information.
- Two-way: Changes in Oracle Internet Directory can be exported to connected directories, and changes in connected directories can be imported into Oracle Internet Directory.

Certain attributes can be targeted or ignored by the synchronization service. For example, the attribute for the employee badge number in Oracle Human Resources may not be of interest to Oracle Internet Directory, its connected directories or client applications. You might not want to synchronize it. On the other hand, the employee identification number may be of interest to those components, so you might want to synchronize it.

Figure 1–2 shows the interactions between components in the Oracle Directory Synchronization Service in a sample deployment.

Figure 1–2 Interactions of the Oracle Directory Synchronization Service



The central mechanism triggering all such synchronization activities is the Oracle Internet Directory change log. It adds one or more entries for every change to any connected directory, including Oracle Internet Directory. The Oracle Directory Synchronization Service:

- Monitors the change log
- Takes action whenever a change corresponds to one or more synchronization profiles
- Supplies the appropriate change to all other connected directories whose individual profiles correspond to the logged change. Such directories could

include, for example, relational databases, Oracle Human Resources, Microsoft Active Directory, or SunONE Directory Server. It supplies these changes using the interface and format required by the connected directory. Synchronization through the Directory Integration and Provisioning connectors ensures that Oracle Internet Directory remains up-to-date with all the information that Oracle Internet Directory clients need.

About the Oracle Provisioning Service

The Oracle Provisioning Service ensures that each provisioned application is notified of changes in, for example, user or group information. To do this, it relies on the information contained in a provisioning integration profile. Each provisioning profile:

- Uniquely identifies the application and organization to which it applies
- Specifies, for example, the users, groups, and operations requiring the application to be notified

The profile must be created when the application is installed, by using the Provisioning Subscription Tool.

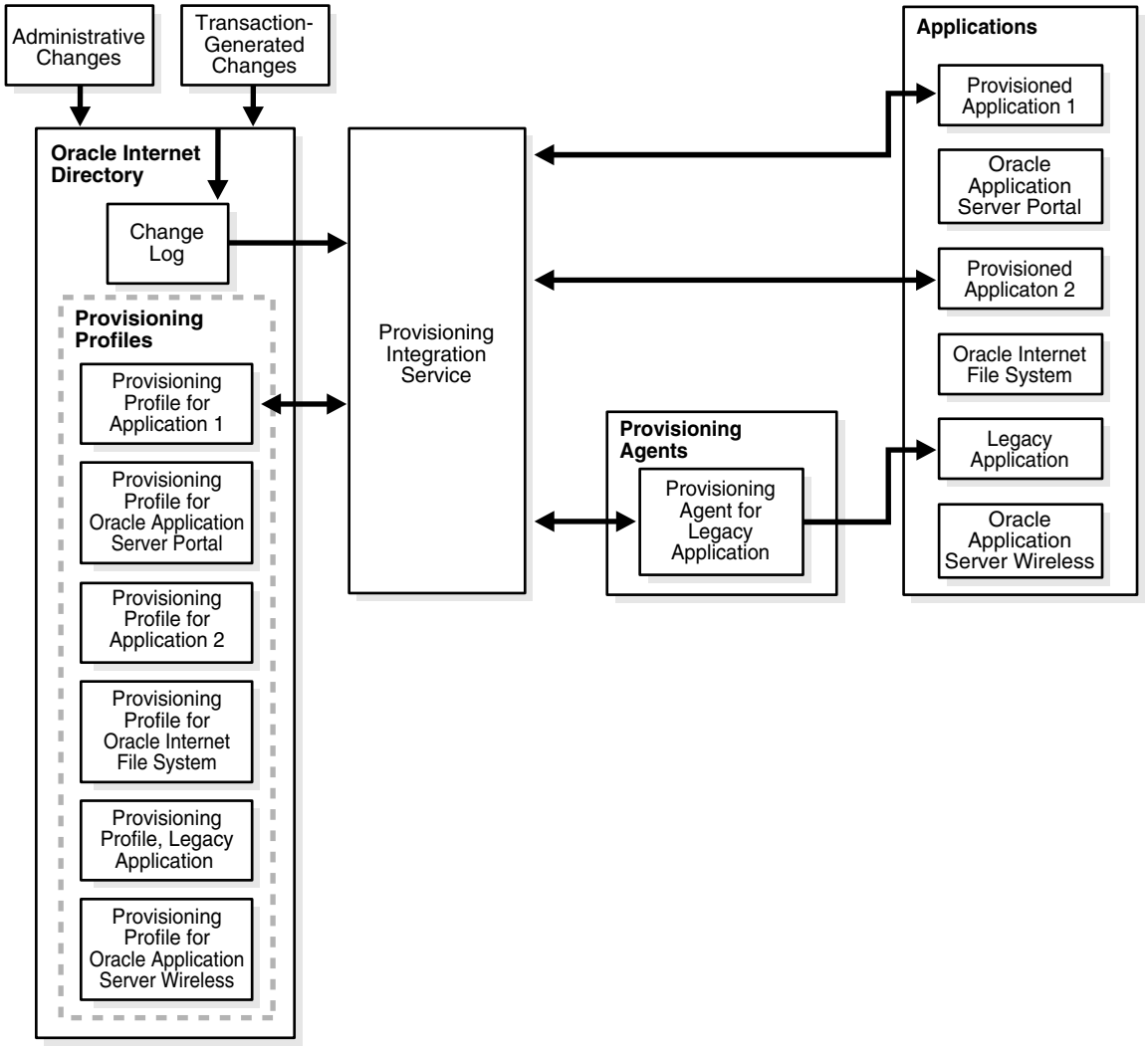
> **See Also:** The chapter on Oracle Directory Integration and Provisioning tools in the *Oracle Identity Management User Reference* for information about the Provisioning Subscription Tool

When changes in Oracle Internet Directory match what is specified in the provisioning profile of an application, the Oracle Provisioning Service sends the relevant data to that application.

Note: A legacy application—that is, one that was operational before the Oracle Provisioning Service was installed—would not have subscribed in the usual way during installation. To enable such an application to receive provisioning information, a **provisioning agent**, in addition to the provisioning profile, must be developed. The agent must be able to translate the relevant data from Oracle Internet Directory into the exact format required by the legacy application.

Figure 1–3 shows the interactions between components in an Oracle Provisioning Service environment, including the special case of a provisioning agent for a legacy application.

Figure 1–3 Interactions of the Oracle Provisioning Service



Oracle Application Server Single Sign-On

Oracle Application Server Single Sign-On enables users to access Oracle Web-based components by logging in only once.

Oracle components delegate the login function to the OracleAS Single Sign-On server. When a user first logs into an Oracle component, the component redirects the login to the OracleAS Single Sign-On server. The OracleAS Single Sign-On server authenticates the user by verifying the credentials entered by the user against those stored in Oracle Internet Directory. After authenticating the user, and throughout the rest of the session, the OracleAS Single Sign-On server grants the user access to all the components the user both seeks and is authorized to use.

> **See Also:** *Oracle Application Server Single Sign-On Administrator's Guide* for information about OracleAS Single Sign-On

2

Security Features in Oracle Directory Integration and Provisioning

This chapter discusses the most important aspects of security in Oracle Directory Integration and Provisioning. It contains these sections:

Authentication in Oracle Directory Integration and Provisioning

Authentication is the process by which the Oracle directory server establishes the true identity of the user connecting to the directory. It occurs when an LDAP session is established by means of the ldapbind operation.

It is important that each component in Oracle Directory Integration and Provisioning be properly authenticated before it is allowed access to the directory.

This section contains these topics:

Secure Sockets Layer (SSL) and Oracle Directory Integration and Provisioning

You can deploy Oracle Directory Integration and Provisioning with or without **Secure Socket Layer (SSL)**. SSL implementation supports these modes:

- No authentication—Provides SSL encryption of data, but does not use SSL for authentication.
- SSL server authentication—Includes both SSL encryption of data and SSL authentication of the server to the client. In Oracle Directory Integration and Provisioning, the server is the directory server, the client is the directory integration and provisioning server.

The server verifies its identity to the client by sending a **certificate** issued by a trusted **certificate authority (CA)**. This mode requires a public key infrastructure (PKI) and SSL wallets to hold the certificates.

To use SSL with Oracle Directory Integration and Provisioning, you must start both the Oracle directory server and Oracle directory integration and provisioning server in the SSL mode.

> **See Also:** The chapter on preliminary tasks and information in Oracle Internet Directory Administrator's Guide for instructions on starting the Oracle directory server in SSL mode

Oracle Directory Integration and Provisioning Server Authentication

You can install and run multiple instances of the directory integration and provisioning server on various hosts. However, when you do this, beware of a malicious user either posing as the directory integration and provisioning server or using an unauthorized copy of it.

To avoid such security issues:

- Ensure that each directory integration and provisioning server is identified properly
- Ensure that, when you start a directory integration and provisioning server, it is properly authenticated before it obtains access to Oracle Internet Directory

Non-SSL Authentication

To use non-SSL authentication, register each directory integration and provisioning server by using the registration tool called odisrvreg.

The registration tool creates:

- An identity entry in the directory. The directory integration and provisioning server uses this entry when it binds to the directory
- An encrypted password. It stores this password in the directory integration and provisioning server entry.
- A private wallet on the local host. This wallet contains the security credentials, including an encrypted password. The name of the wallet is specified in the `odi.properties` file and it is stored in the $*ORACLE_HOME*/ldap/odi/conf directory.

When it binds to the directory, the directory integration and provisioning server uses the encrypted password in the private wallet.

Note: Ensure that the wallet is protected against unauthorized access.

> **See Also:** "Manually Registering the Oracle Directory Integration and Provisioning Server" on page 4-14 for instructions on registering the directory integration and provisioning server

Authentication in SSL Mode

The identity of the directory server can be established by starting both Oracle Internet Directory and the directory integration and provisioning server in the SSL server authentication mode. In this case, the directory server provides its certificate to the

directory integration and provisioning server, which acts as the client of Oracle Internet Directory.

The directory integration and provisioning server is authenticated by using the same mechanism used in the non-SSL mode.

You can also configure the Oracle directory integration and provisioning server to use SSL when connecting to a third-party directory. In this case, you store the connected directory certificates in the wallet as described in "Managing the SSL Certificates of Oracle Internet Directory and Connected Directories" on page 4-7.

Profile Authentication

Within Oracle Internet Directory, an integration profile represents a user with its own DN and password. The users who can access the profiles are:

- The administrator of Oracle Directory Integration and Provisioning (`DIPAdmin`), represented by the DN `cn=dipadmin,cn=odi,cn=oracle internet directory`
- Members of the Oracle Directory Integration and Provisioning administrator group (`DIPAdminGroup`), represented by the DN `cn=dipadmingroup,cn=odi,cn=oracle internet directory`

When the directory integration and provisioning server imports data to Oracle Internet Directory based on an integration profile, it proxy-binds to the directory as that integration profile.The Oracle directory integration and provisioning server can bind in either SSL and non-SSL mode.

Access Control and Authorization and Oracle Directory Integration and Provisioning

Authorization is the process of ensuring that a user reads or updates only the information for which that user has privileges. When directory operations are attempted within a directory session, the directory server ensures that the user—identified by the authorization identifier associated with the session—has the requisite permissions to perform those operations. If the user does not have the necessary permissions, then the directory server disallows the operation. Through this mechanism, called access control, the directory server protects directory data from unauthorized operations by directory users.

To restrict access to only the desired subset of Oracle Internet Directory data, for both the directory integration and provisioning server and a connector, place appropriate access policies in the directory.

This section discusses these policies in detail. It contains these topics:

- Access Controls for the Oracle Directory Integration and Provisioning Server
- Access Controls for Profiles

Access Controls for the Oracle Directory Integration and Provisioning Server

The directory integration and provisioning server binds to the directory both as itself and on behalf of the profile.

- When it binds as itself, it can cache the information in various integration profiles. This enables the directory integration and provisioning server to schedule synchronization actions to be carried out by various connectors.

- When the directory integration and provisioning server operates on behalf of a profile, it proxies as the profile—that is, it uses the profile credentials to bind to the directory and perform various operations. The directory integration and provisioning server can perform only those operations in the directory that are permitted to the profile.

To establish and manage access rights granted to directory integration and provisioning servers, Oracle Directory Integration and Provisioning creates a group entry, called `odisgroup`, during installation. The DN of `odisgroup` is `cn=odisgroup,cn=odi,cn=oracle internet directory`. When a directory integration and provisioning server is registered, it becomes a member of this group.

You control the access rights granted to directory integration and provisioning servers by placing access control policies in the `odisgroup` entry. The default policy grants various rights to directory integration and provisioning servers for accessing the profiles. For example, the default policy enables the directory integration and provisioning server to compare user passwords between Oracle Internet Directory and a connected directory it binds as proxy on behalf of a profile. It also enables directory integration and provisioning servers to modify status information in the profile—such as the last successful execution time and the synchronization status.

Access Controls for Profiles

To control access to Oracle Internet Directory data by integration profiles, place appropriate access control policies in Oracle Internet Directory. This enables you to protect data synchronized or processed by one profile from interference by another profile. It also enables you to allow only the integration profile that owns synchronization of an attribute to modify that attribute.

> **See Also:** The chapter on access control, specifically, the section security groups, in *Oracle Internet Directory Administrator's Guide* for instructions on setting access control policies for group entries.

For example, creating a group entry called `odipgroup` when installing the Oracle Internet Directory enables you to control the access rights granted to various profiles. Rights are controlled by placing appropriate access policies in the `odipgroup` entry. Each profile is a member of this group. The membership is established when the profile is registered in the system. The default access policy, automatically installed with the product, grants to profiles certain standard access rights for the integration profiles they own. One such right is the ability to modify status information in the integration profile, such as the parameter named `orclodipConDirLastAppliedChgTime`. The default access policy also permits profiles to access Oracle Internet Directory change logs, to which access is otherwise restricted.

The `odisgroup` group entries and their default policies are created during the server installation of the Oracle Internet Directory. Oracle Directory Integration and Provisioning-only installations do not create these groups and policies.

Data Integrity and Oracle Directory Integration and Provisioning

Oracle Directory Integration and Provisioning ensures that data has not been modified, deleted, or replayed during transmission by using SSL. This SSL feature generates a cryptographically secure message digest—through cryptographic checksums using either the MD5 algorithm or the Secure Hash Algorithm (SHA)—and includes it with each packet sent across the network.

Data Privacy and Oracle Directory Integration and Provisioning

Oracle Directory Integration and Provisioning ensures that data is not disclosed during transmission by using public-key encryption available with SSL. In public-key encryption, the sender of a message encrypts the message with the public key of the recipient. Upon delivery, the recipient decrypts the message using the recipient's private key.

To exchange data securely between the directory integration and provisioning server and Oracle Internet Directory, you run both components in the SSL mode.

Tools Security and Oracle Directory Integration and Provisioning

You can run all the commonly used tools in the SSL mode to transmit data to Oracle Internet Directory securely. These tools include:

- Oracle Directory Manager —Use it to administer data in the directory.
- The Oracle directory integration and provisioning server registration tool (`odisrvreg`)—Use it to register the directory integration and provisioning server in the directory.
- The Oracle Directory Integration and Provisioning Server Administration tool
- The Directory Integration and Provisioning Assistant when running in SSL mode
- The Provisioning Subscription Tool when running in the SSL mode

Part II

General Administration of Oracle Directory Integration and Provisioning

This section of the *Oracle Identity Management Integration Guide* describes some of the more general administrative tasks involved in running Oracle Directory Integration and Provisioning. You can find more specific administrative information in the respective sections of the book.

Part II contains the following chapters:

3

Oracle Directory Integration and Provisioning Administration Tools

This chapter describes the Oracle Directory Integration and Provisioning Server Administration tool along with various other tools used for administering Oracle Directory Integration and Provisioning. It contains these topics:

- The Oracle Directory Integration and Provisioning Server Administration Tool
- Graphical Tools for Oracle Directory Integration and Provisioning Administration
- Command-Line Tools for Oracle Directory Integration and Provisioning Administration

The Oracle Directory Integration and Provisioning Server Administration Tool

The Oracle Directory Integration and Provisioning Server Administration tool is a Java-based utility for graphically administering the Oracle directory integration and provisioning server. This section describes some of its basic features. More specific instructions are found in sections throughout this book that explain how to perform various tasks.

This section contains these topics:

- Starting the Oracle Directory Integration and Provisioning Server Administration Tool
- Connecting to a Directory Server by Using the Oracle Directory Integration and Provisioning Server Administration Tool
- Navigating the Oracle Directory Integration and Provisioning Server Administration Tool
- Disconnecting from a Directory Server by Using the Oracle Directory Integration and Provisioning Server Administration Tool
- Configuring the Display and Duration of Searches in the Oracle Directory Integration and Provisioning Server Administration Tool
- Configuring the Display of ACPs in the Oracle Directory Integration and Provisioning Server Administration Tool

Starting the Oracle Directory Integration and Provisioning Server Administration Tool

Before you can launch the Oracle Directory Integration and Provisioning Server Administration tool, you must have a directory server instance running.

> **See Also:** Chapter 7, "Administration of Directory Synchronization" for information on the Oracle Directory Integration and Provisioning Server Administration tool

To start the Oracle Directory Integration and Provisioning Server Administration tool, follow the instructions for your operating system as described in Table 3–1:

Table 3–1 Operating System-Specific Instructions for Starting Oracle Directory Integration and Provisioning Server Administration tool

Operating System	Instructions
Windows NT	From the **Start** menu, choose **Programs**, then ***ORACLE_HOME***, then **Integrated Management**, then **Oracle Directory Integration and Provisioning Server Administration**
UNIX	If you have not set the path, then navigate to $*ORACLE_HOME*/bin. At the system prompt, enter: `dipassistant -gui`

The first time you start the Oracle Directory Integration and Provisioning Server Administration tool, an alert tells you that you must connect to a server. Choose **OK**. The Directory Server Connection dialog box appears.

Connecting to a Directory Server by Using the Oracle Directory Integration and Provisioning Server Administration Tool

> **Note:** To use this tool, you must be a member of the following group: `cn=dipadmingrp,cn=odi,cn=oracle internet directory`. If you do not have the correct privileges, then access to the tool is denied.

To connect to a directory server:

1. In the Directory Server Connection dialog box, type the name and port number of an available server.

   The default port is 389. You can change the port if you wish. However, if you have an Oracle directory server running on a port that is not the default, then be sure that any clients that use that server are informed of the correct port.

   Choose **OK**. The Oracle Directory Integration and Provisioning Server Administration Connect dialog box appears.

   If the directory server to which you want to connect does not appear in the initial login window—that is, it is not the default directory server—then you can select another directory server by clicking the button to the right of the Server field.

   This dialog box then displays a list of all directory servers to which you have connected at any time in the past. You can select a directory server from the list, either to connect to it, delete it, edit it, or to use it as a template for another management connection.

   To connect to a server from the list, select it and choose Select at the bottom of the dialog box. The server and port appear in the Oracle Internet Directory Connect dialog box, from which you can connect.

To delete an existing defined connection, select the server, then choose Delete. The server entry is removed from your list of defined management connections.

To define a new management connection:

- To add a new management connection, choose Add. This displays the Directory Server Connection dialog box. After you enter a server name and port in this dialog box and choose OK, the new management connection appears in the list in the Select Directory Server dialog box. From here you can select it to appear in the Oracle Internet Directory Connect dialog box, and thus connect.
- To use an existing management connection as the template for a new connection, select the server you want to use as a template, then click Add Like. The Directory Server Connection dialog box appears, with the template server information filled in. You must edit these entries to create a new management connection. After you enter a server name and port in this dialog box and click OK, the new management connection appears in the list in the Select Directory Server dialog box. From here you can select it to appear in the Oracle Internet Directory Connect dialog box, and thus connect.
- To edit an existing connection, select it, then click Edit. The Directory Server Connection dialog box appears, with the server and port information filled in. Edit the entries and save any changes. After you enter a server name and port in this dialog box and click OK, the new management connection appears in the list in the Select Directory Server dialog box. From here you can select it to appear in the Oracle Internet Directory Connect dialog box, and thus connect.

2. In each field of the Credentials tab page, type the information specific to this server instance.

   The fields in the Credentials tab page are described in Table A–1 on page A-2.

   See Also:

   - The chapter on SSL and the directory in *Oracle Internet Directory Administrator's Guide* for instructions on enabling SSL and on the impact of changing ports on security
   - The section about entries in the concepts chapter of *Oracle Internet Directory Administrator's Guide* for instructions on formatting distinguished names
   - *Oracle Advanced Security Administrator's Guide* for instructions on creating a wallet by using Oracle Wallet Manager when using SSL

3. If you selected the **SSL Enabled** check box on the **Credentials** tab page, then select the **SSL** tab.
4. In the SSL tab page, enter the requested data in the fields.

   The fields in the SSL tab page are described in Table A–2 on page A-4
5. Choose **Login**. The Oracle Directory Integration and Provisioning Server Administration tool appears.

Navigating the Oracle Directory Integration and Provisioning Server Administration Tool

This section provides an overview of Oracle Directory Integration and Provisioning Server Administration, and explains the items in the menu bar and the buttons on the toolbar.

Overview of Oracle Directory Integration and Provisioning Server Administration

Like the directory itself, the navigator pane (left side of the double window interface) has a tree-like structure. When the tool first opens, the navigator pane shows only one tree item. By clicking the plus sign(+) next to the tree item, subcomponents of that tree item appear.

In the right pane, some windows contain buttons labeled Apply and OK. If you choose Apply, then your changes are committed, and the window remains available for more changes. If you choose OK, then your changes are committed, and the window closes.

Similarly, some windows have buttons that are labeled Revert and Cancel. If you press Revert, then your changes in that window do not take effect, the original values reappear in the fields, and the window stays open for further work. If you press Cancel, then your changes in that window do not take effect, and the window closes.

The Oracle Directory Integration and Provisioning Server Administration Menu Bar

Table 3–2 lists and describes the menus you can access by using the menu bar. Menu items become enabled or disabled depending on the pane or tab page you are displaying.

Table 3–2 Oracle Directory Integration and Provisioning Server Administration Menu Bar

Menu	Menu Items
File	**Create**—Adds an object
	Create Like—Adds a new object by using the object selected in the navigator pane as a template
	Connect—Connects to a directory server selected in the navigator pane
	Disconnect—Disconnects from a directory server selected in the navigator pane
	Exit—Exits the Oracle Directory Integration and Provisioning Server Administration tool
Edit	**Edit**—Modifies an object
	Remove—Removes a selected object
	Find Objects—Searches for either an object class or an attribute, depending on the context.
View	**Refresh**—Updates data stored in memory to reflect changes in the database
	Tear-Off—Generates a secondary dialog containing the fields and values displayed in the Oracle Directory Integration and Provisioning Server Administration tool's right pane. This is useful when comparing two pieces of information.
Help	**Contents**—Displays the Contents tab page of the Help navigator
	Search for Help On...—Displays the Help Search dialog box that you use to search for words in the online help guide
	About Oracle Internet Directory—Displays Oracle Internet Directory version information

Disconnecting from a Directory Server by Using the Oracle Directory Integration and Provisioning Server Administration Tool

To disconnect from a directory server by using the Oracle Directory Integration and Provisioning Server Administration tool, from the **File** menu choose **Disconnect**. Also, when you exit the Oracle Directory Integration and Provisioning Server Administration tool, connections between all directory servers and the directory are automatically disconnected.

All connection information is stored in the user's home directory in the file `osdadmin.ini`.

When you restart the Oracle Directory Integration and Provisioning Server Administration tool, all previously connected server connections appear in the Directory Server Login dialog box.

Configuring the Display and Duration of Searches in the Oracle Directory Integration and Provisioning Server Administration Tool

You can specify the maximum number of entries to be displayed in the Oracle Directory Integration and Provisioning Server Administration tool as the result of searches and the duration of searches. You can make these configurations in either this tool or the directory server or both.

If you make the configuration in both this tool and the directory server, and the two configurations do not match, then Oracle Internet Directory resolves the conflict as follows:

- If the value you set in this tool is greater than that in the directory server, then the configuration of the server prevails. For example, if you set this tool to search for 2 minutes, and the directory server for 3 minutes, then the actual search duration will be 3 minutes.
- If the value you set in this tool is less than that in the directory server, then the configuration of this tool prevails. For example, if you set this tool to search for 2 minutes, and the server for 3 minutes, then the actual search duration is 2 minutes.

To configure the display and duration of searches in the Oracle Directory Integration and Provisioning Server Administration tool:

1. In the navigator pane, select the server you want to configure.
2. From the toolbar, select **User Preferences**. The User Preferences dialog box appears.
3. In the Configure Entry Management tab page, in the field labeled **Maximum number of one-level subtree entries**, enter the maximum number of entries to be returned by a search. The default is 200.
4. In the **Search Time Limit** field, enter the maximum number of seconds for a search to be completed. The default is 25.
5. Choose **OK**.

> **See Also:** "Configure Entry Management" on page A-4 for more information about this tab page.

Configuring the Display of ACPs in the Oracle Directory Integration and Provisioning Server Administration Tool

The Oracle Directory Integration and Provisioning Server Administration tool enables you to determine whether the navigator pane displays all ACPs automatically or only as the result of a search. If you have a large number of ACPs, then you may want to display them only as the result of a search.

To configure the display of ACPs:

1. In the navigator pane, select the server you want to configure.
2. On the toolbar, choose **User Preferences**. The User Preferences dialog box appears.
3. Select the Configure Access Control Policy Management tab page.
4. Select either:
   - **Always display all ACPs**
   - **Only display ACPs based on search request**
5. Choose **OK**.
6. To effect your changes, restart the Oracle Directory Integration and Provisioning Server Administration tool.

Graphical Tools for Oracle Directory Integration and Provisioning Administration

In addition to the Oracle Directory Integration and Provisioning Server Administration tool, you can use the following graphical tools to administer Oracle Directory Integration and Provisioning:

- Oracle Directory Manager
- Oracle Internet Directory Self-Service Console
- Oracle Internet Directory Provisioning Console

Oracle Directory Manager

Oracle Directory Manager is a Java-based tool for graphically administering Oracle Internet Directory. You can use Oracle Directory Manager to:

- Create, modify, and delete directory integration profiles for synchronization
- Monitor synchronization profiles and synchronization status
- Monitor the status of all Oracle directory integration and provisioning server instances
- Troubleshoot synchronization

See Also:

- *Oracle Internet Directory Administrator's Guide* for more information on Oracle Directory Manager
- Chapter 4, "Managing the Oracle Directory Integration and Provisioning Server"

Oracle Internet Directory Self-Service Console

The Oracle Internet Directory Self-Service Console enables you to delegate administrative privileges to various administrators and to end users. It is a ready-to-use standalone application created by using Oracle Delegated Administration Services that provides a single graphical interface for delegated administrators and end users to manage data in the directory. The Oracle Internet Directory Self-Service Console enables both administrators and end users, depending on their privileges, to perform various directory operations. In an integrated deployment, the Oracle Internet Directory Self-Service Console is primarily used for customizing realm parameters.

> **See Also:** The Oracle Internet Directory Self-Service Console chapter in *Oracle Identity Management Guide to Delegated Administration*.

Oracle Internet Directory Provisioning Console

The Oracle Internet Directory Provisioning Console provides a single graphical interface for administrators to provision users in Oracle Internet Directory. The Provisioning Console was created with Oracle Delegated Administration Services, and works alongside the Oracle Internet Directory Self-Service Console.

> **See Also:** Part IV, "Provisioning in Oracle Identity Management"

Command-Line Tools for Oracle Directory Integration and Provisioning Administration

The following command-line tools are available for administering Oracle Directory Integration and Provisioning:

- OID Control and OID Monitor
- The Oracle Directory Integration and Provisioning Server Registration Tool (odisrvreg)
- Directory Integration and Provisioning Assistant (dipassistant)
- The Provisioning Subscription Tool (oidprovtool)
- Entry and Attribute Management Command-Line Tools
- The schemasync Tool

> **See Also:** *Oracle Identity Management User Reference* for the required syntax for each of the tools discussed in this section, along with information on other command-line tools that you can use to administer Oracle Internet Directory and Oracle Directory Integration and Provisioning

OID Control and OID Monitor

OID Control and OID Monitor enable you to start, stop, and monitor the Oracle directory integration and provisioning server.

In Oracle Internet Directory, you can use OID Control and OID Monitor to control the directory integration and provisioning server in the `ORACLE_HOME` where either the Oracle directory server or Oracle directory integration and provisioning server are installed.

If the Oracle Internet Directory installation is client-only, then the OID Control Utility and OID Monitor are not installed. In this case, start Oracle directory integration and provisioning server manually. In this configuration you can still use Oracle Directory Integration and Provisioning Server Administration tool to learn the status of Oracle directory integration and provisioning server.

> **See Also:** Chapter 4, "Managing the Oracle Directory Integration and Provisioning Server"

The Oracle Directory Integration and Provisioning Server Registration Tool (odisrvreg)

The Oracle Directory Integration and Provisioning Server Registration tool (`odisrvreg`) registers an Oracle directory integration and provisioning server with the directory. It does this by creating an entry in the directory and setting the password for the Oracle directory integration and provisioning server. If the registration entry already exists, then you can use the `odisrvreg` tool to reset the existing password. The `odisrvreg` tool also creates a local file named odisrvwallet_hostname, at $*ORACLE_HOME*/ldap/odi/conf. This file acts as a private wallet for the Oracle directory integration and provisioning server, which uses it on startup to bind to the directory.

Directory Integration and Provisioning Assistant (dipassistant)

The Directory Integration and Provisioning Assistant (`dipassistant`) is the command-line version of the Oracle Directory Integration and Provisioning Server Administration tool. Some of the tasks you can perform with the Directory Integration and Provisioning Assistant include:

- Creating, modifying, and deleting synchronization profiles
- Viewing all synchronization profile names in Oracle Internet Directory
- Viewing the details of a specific synchronization profile
- Migrating data (or "bootstrapping") between a connected directory and Oracle Internet Directory
- Setting the wallet password for Oracle directory integration and provisioning server
- Resetting the password of the Oracle Directory Integration and Provisioning administrator
- Moving integration profiles to a different Oracle Internet Directory node

The Provisioning Subscription Tool (oidprovtool)

You use the Provisioning Subscription tool (`oidprovtool`) to administer provisioning profile entries in the directory. More specifically, you can use Provisioning Subscription tool to:

- Create new provisioning profiles
- Enable/disable existing provisioning profiles
- Modify existing provisioning profiles
- Delete existing provisioning profiles
- Get the current status of a provisioning profile
- Clear all errors in an existing provisioning profile

Entry and Attribute Management Command-Line Tools

Table 3–3 lists the entry and attribute management command-line tools that you can use with Oracle Directory Integration and Provisioning.

Table 3–3 Entry and Attribute Management Command-Line Tools

Tool	Description
Catalog Management Tool (`catalog.sh`)	Indexes attributes
`ldapadd`	Add entries and their object classes, attributes, and values to the directory
`ldapaddmt`	Supports multiple threads for concurrently adding entries and their object classes, attributes, and values to the directory
`ldapbind`	Determines whether you can authenticate a client to a server
`ldapcompare`	Matches specified attribute values with an entry's attribute values
`ldapdelete`	Removes entries from the directory
`ldapmoddn`	Modifies an entry's DN or RDN
`ldapmodify`	Modifies an entry's attributes
`ldapmodifymt`	Supports multiple threads for modifying entries concurrently
`ldapsearch`	Searches for entries in the directory

The schemasync Tool

The `schemasync` tool enables you to synchronize schema elements—namely attributes and object classes—between Oracle Internet Directory and third-party LDAP directories.

4

Managing the Oracle Directory Integration and Provisioning Server

This chapter discusses the Oracle directory integration and provisioning server and explains how to configure and manage it. It contains these topics:

- Operational Information about the Oracle Directory Integration and Provisioning Server
- Viewing Oracle Directory Integration and Provisioning Server Information
- Managing Configuration Set Entries Used by the Oracle Directory Integration and Provisioning Server
- Managing the SSL Certificates of Oracle Internet Directory and Connected Directories
- Starting, Stopping, and Restarting the Oracle Directory Integration and Provisioning Server
- Starting and Stopping the Oracle Directory Integration and Provisioning Server in a High Availability Scenario
- Setting the Debug Level for the Oracle Directory Integration and Provisioning Server
- Managing Oracle Directory Integration and Provisioning in a Replicated Environment
- Finding the Log Files
- Manually Registering the Oracle Directory Integration and Provisioning Server

> **See Also:** "Oracle Directory Integration and Provisioning Server" on page 1-5 for a summary of the functions performed by the Oracle directory integration and provisioning server

---

Note: For security reasons, Oracle Corporation recommends that you run the Oracle directory integration and provisioning server on the same host as the directory server. If you run them on different hosts, then run them by using SSL as described in the chapter on SSL and the directory in *Oracle Internet Directory Administrator's Guide*.

Operational Information about the Oracle Directory Integration and Provisioning Server

This section introduces structural and operational information about the directory integration and provisioning server and contains these topics:

Directory Integration Profiles

In Oracle Directory Integration and Provisioning, you can create two types of profiles: a directory synchronization profile and a directory provisioning profile. A **directory synchronization profile** describes how synchronization is carried out between Oracle Internet Directory and an external system. You can create two types of directory synchronization profiles: an import profile and an export profile. An import profile imports changes from a connected directory to Oracle Internet Directory while an export profiles exports changes from Oracle Internet Directory to a connected directory. A **directory provisioning profile** describes the nature of provisioning-related notifications that Oracle Directory Integration and Provisioning sends to the directory-enabled applications. Each type of profiles is special kind of **directory integration profile**, which is an entry in Oracle Internet Directory that describes how Oracle Directory Integration and Provisioning communicates with external systems and what is communicated.

The Oracle Directory Integration and Provisioning Server and Configuration Set Entries

Each directory integration and provisioning server can execute a set of connectors either for:

- Synchronizing between Oracle Internet Directory and connected directories. The set of connectors for synchronization is provided in the configuration set number entered in the command line when starting the Oracle directory integration and provisioning server.
- Provisioning users, groups, and realms for Oracle components. The set of profiles for provisioning is provided in the `grpID` argument in the command line when starting the Oracle directory integration and provisioning server.

If the configuration set number is not specified, then the directory integration and provisioning server starts in the mode for processing provisioning profiles. If the configuration set number is specified, but there are no integration profiles in the directory for the specified configuration set number, then the directory integration and provisioning server waits indefinitely until integration profiles are added to that configuration set. This wait also occurs if integration profiles are configured for the configuration set but disabled.

If the configuration set specified in the command line does not exist in the directory, then the directory integration and provisioning server logs this information in the log file and exits. For provisioning profiles, the same behavior is followed for the `grpID` attribute, which is passed as an argument in the command line.

Whenever a connector is scheduled to do synchronization or provisioning, the directory integration and provisioning server starts a separate thread. This thread opens an LDAP connection to the directory server to read or write entries from Oracle Internet Directory, and then closes the connection before exiting.

The directory integration and provisioning server executes three types of threads in the process, and these are described in Table 4–1:

Table 4–1 Oracle Directory Integration and Provisioning Server Threads

Thread	Description
Main thread	Daemon thread of the Oracle directory integration and provisioning server. To look for changed profiles and to refresh its cache, it starts up the scheduler and periodically sends refresh signals to it. This thread also looks for the shutdown signal from the OID Monitor (`oidmon`). This signal causes the thread to shut itself down after it sends a signal to the scheduler to shut down.
Scheduler thread	Scheduler for the connectors for synchronization based on their specified scheduling interval. On receipt of a refresh signal from the main thread, this thread refreshes the synchronization profiles to the latest values.
Connector thread	In a synchronization, the thread that invokes the connector executable named in the profile, and maps and filters the attributes. It is spawned by the scheduler at the specified individual scheduling intervals. Once all the changes from the source directory are propagated to the destination directory, this thread exits.

Standard Sequences of Directory Integration and Provisioning Server Events

Each instance of the Oracle directory integration and provisioning server supports either provisioning or synchronization. The directory integration and provisioning server runs as a shared server process while handling the synchronization and provisioning event propagations.

The three threads described in Table 4–1 on page 4-3 work together to create these typical process flow sequences:

Main Thread Process Sequence

On startup, the main thread comes up. This daemon thread of the server starts the scheduler. It verifies the registration of the instance in the directory. If the instance is not registered, then it is not started up by OID Monitor. Instead, it registers itself in Oracle Internet Directory with the configuration set number and the instance number details.

The main thread periodically checks for the refresh time and signals the scheduler to refresh. It also periodically checks for the shutdown signal. On receipt of the shutdown signal, it signals the scheduler thread to shutdown.

Once the scheduler thread shuts down, the main thread unregisters and shuts down.

Scheduler Thread Process Sequence

When it is started by the main thread, the scheduler thread reads the configuration set to determine which integration profiles to schedule. It creates a list of profiles to be scheduled and schedules them based on their specified scheduling interval. While creating the list of profiles, it validates the attributes. If any of the profile attributes have invalid values, the profile is not considered for synchronization or provisioning.

When it receives the refresh signal, the scheduler thread refreshes the integration profiles. When it receives the shutdown signal, the scheduler thread waits until all the connectors complete the synchronization or provisioning event propagation. It then returns control to the main thread.

Connector Thread Process Sequence for Synchronization

A synchronization thread follows this process:

1. Establishes connection with the connected directory and Oracle Internet Directory
2. In an import operation, executes any agent execution command that may be specified in the connector
3. Opens the DB/LDAP/LDIF/Tagged file if required
4. Reads the changes from the source one at a time
5. Filters the changes if applicable
6. Maps the changes as specified by the mapping rules
7. Creates the destination change record
8. Write the changes to the destination
9. After applying all the changes, closes the thread

Connector Thread Process Sequence for Provisioning

A provisioning thread follows this process:

1. Establishes a connection with the connected directory
2. Reads the changes from the source, one at a time
3. Filters the changes if applicable
4. Identifies the change as a specific event—that is:
   - USER Add/Modify/Delete
   - GROUP Add/Modify/Delete
5. Creates the event notification record
6. Invokes the given package to consume the event notification

Oracle Directory Integration and Provisioning Event Propagation in a Multimaster Oracle Internet Directory Replication Environment

In a multimaster Oracle Internet Directory replication environment, changes to directory integration profiles on one Oracle Internet Directory node are not automatically replicated on other Oracle Internet Directory nodes. For this reason, you must observe the considerations that are outlined in this section when implementing Oracle Directory Integration and Provisioning in a multimaster Oracle Internet Directory replication environment.

Directory Synchronization in a Multimaster Oracle Internet Directory Replication Environment

Because directory synchronization profiles on a primary Oracle Internet Directory node are not automatically replicated to secondary Oracle Internet Directory nodes, you should manually copy the profiles on the primary node to any secondary nodes on a periodic basis. This allows a directory synchronization profile to execute on a

secondary node in the event of a problem on the primary node. However, the value assigned to the `lastchangenumber` attribute in a directory synchronization profile is local to the Oracle Internet Directory node where the profile is located. This means that if you simply copy a directory synchronization profile from one Oracle Internet Directory node to another, the correct state of synchronization or event propagation will not be preserved.

Note: If the primary node running either the directory replication server (`oidrepld`), or the Oracle directory integration and provisioning server (`odisrv`), or both fails, then the OID Monitor on the secondary node starts these processes on the secondary node after five minutes. However, when the primary node is restarted, these servers are not automatically restarted on the primary node.

Normal shutdown is not treated as a failover—that is, after a normal shutdown, the OID Monitor on the secondary node does not start these processes on the secondary node after five minutes. However, as in the case of a failure, when the primary node is restarted, these servers are not automatically restarted on the primary node.

When copying import profiles from one node to another, the `lastchangenumber` attribute is irrelevant because the value is obtained from the connected directory. However, after copying an export profile to a target node, you must update the `lastchangenumber` attribute with the value from the target node as follows:

1. Stop the Oracle directory integration and provisioning server as explained in "Stopping the Oracle Directory Integration and Provisioning Server" on page 4-9.
2. Obtain the value of the `lastchangenumber` attribute on the target node by following the instructions in the `dipassistant showprofile` section in the Oracle Directory Integration and Provisioning tools chapter of the *Oracle Identity Management User Reference*.
3. Copy the directory synchronization profiles from the primary node to the target nodes by following the instructions in the `dipassistant reassociate` section of the Oracle Directory Integration and Provisioning tools chapter of of the *Oracle Identity Management User Reference*.
4. Use the Oracle Directory Integration and Provisioning Server Administration tool or the Directory Integration and Provisioning Assistant (`dipassistant`) to update the `lastchangenumber` attribute in the export profile you copied to the target node with the value you obtained in Step 2.

   See Also:

   - "The Oracle Directory Integration and Provisioning Server Administration Tool" on page 3-1
   - *Oracle Identity Management User Reference*

5. Start the Oracle directory integration and provisioning server as explained in "Starting the Oracle Directory Integration and Provisioning Server" on page 4-8.

Directory Provisioning in a Multimaster Oracle Internet Directory Replication Environment

In a default multimaster Oracle Internet Directory replication environment, the Oracle directory integration and provisioning server is installed in the same location as the

primary Oracle Internet Directory. If the primary node fails, event propagation stops for all profiles located on the node. Although the events are queued and not lost while the primary node is stopped, the events will not be propagated to any applications that expect them. In order to ensure that events continue to be propagated even when the primary node is down, you must copy the directory provisioning profiles to other secondary nodes in a multimaster Oracle Internet Directory environment. However, directory provisioning profiles should only be copied from the primary node to any secondary nodes immediately after an application is installed and before any user changes are made in Oracle Internet Directory.

To copy the directory provisioning profiles from a primary node to any secondary nodes, follow the instructions in the `dipassistant reassociate` command section in the Oracle Directory Integration and Provisioning tools chapter of the *Oracle Identity Management User Reference*.

Viewing Oracle Directory Integration and Provisioning Server Information

When the directory integration and provisioning server starts, it generates specific runtime information and stores it in the directory. This information includes:

- The instance number of the directory integration and provisioning server
- The host on which it is running
- The configuration set with which the directory integration and provisioning server was started
- The group identifier of the provisioning profile group it is running

You can view this information by using either the Oracle Directory Integration and Provisioning Server Administration tool or the `ldapsearch` utility, as described in the following topics:

- Viewing Oracle Directory Integration and Provisioning Server Runtime Information by Using the Oracle Directory Integration and Provisioning Server Administration Tool
- Viewing Oracle Directory Integration and Provisioning Server Runtime Information by Using ldapsearch

Viewing Oracle Directory Integration and Provisioning Server Runtime Information by Using the Oracle Directory Integration and Provisioning Server Administration Tool

To view runtime information for the directory integration and provisioning server instance by using the Oracle Directory Integration and Provisioning Server Administration tool:

1. In the navigator pane, expand the ***directory server instance***.
2. Select **Integration Profile Configuration**. The Active Processes box appears in the right pane and displays the Oracle directory integration and provisioning server runtime information.

Viewing Oracle Directory Integration and Provisioning Server Runtime Information by Using ldapsearch

To view registration information for the directory integration and provisioning server instance by using the `ldapsearch` utility, perform a base search on its entry. For example:

```
ldapsearch -p 389 -h my_host -b cn=instance1,cn=odisrv,cn=subregistrysubentry -s
base -v "objectclass=*"
```

This example search returns the following:

```
dn: cn=instance1,cn=odisrv,cn= subregistrysubentry
cn: instance1
orclodipconfigdns: orclodipagentname=HRAgent,cn=subscriber profile,cn=changelog
subscriber,cn=oracle internet directory
orcldiaconfigrefreshflag: 0
orclhostname: my_host
orclconfigsetnumber: 1
objectclass: top
objectclass: orclODISInstance
```

Managing Configuration Set Entries Used by the Oracle Directory Integration and Provisioning Server

You can create, modify, and view configuration set entries by using either the Oracle Directory Integration and Provisioning Server Administration tool or the Directory Integration and Provisioning Assistant. When a connector is registered, an integration profile is created and added to the given configuration set. This configuration set entry determines the behavior of the directory integration and provisioning server.

You can control the runtime behavior of the directory integration and provisioning server by using a different configuration set entry when you start it. For example, you can start instance 1 of the directory integration and provisioning server on host H1 with configset1, and instance 2 on host H1 with configset2. The behavior of instance 1 depends on configset1, and that of instance 2 depends on configset2. Dividing the agents on host H1 between two configuration set entries distributes the load between the two directory integration and provisioning server instances. Similarly, running different configuration sets and different instances on different hosts balances the load between the servers.

Managing the SSL Certificates of Oracle Internet Directory and Connected Directories

The Oracle directory integration and provisioning server can use SSL to connect to Oracle Internet Directory and connected directories. When using SSL with no authentication to connect to Oracle Internet Directory, no certificate is required. However, when connecting to connect to Oracle Internet Directory using SSL with server authentication, you need a trust point certificate to connect to the LDAP server. The Oracle directory integration and provisioning server expects the certificate to be in a wallet, which is a data structure used to store and manage security credentials for an individual entity. Oracle Wallet Manager is an application that wallet owners and security administrators use to manage and edit the security credentials in their wallets.

> **See Also:** The chapter on Oracle Wallet Manager in *Oracle Advanced Security Administrator's Guide*

The location of the wallet and the password to open it are stored in a properties file used by Directory Integration and Provisioning. This file is *$ORACLE_HOME*/ldap/odi/conf/odi.properties.

A typical `odi.properties` file has the entries described in Table 4–2. You must update the `odi.properties` file with values that are appropriate to your deployment.

Table 4–2 Entries in the odi.properties File

Entry	Description
`RegWalletFile: odi/conf/srvWallet`	This entry indicates the location of the registration information of Directory Integration and Provisioning with Oracle Internet Directory. The location of the file is in relation to the *$ORACLE_HOME*/ldap directory.
`CertWalletFile: location_of_certificate_wallet`	Location of the certificate wallet. The certificate wallet file is the location of the `ewallet.p12` file.
`CertWalletPwdFile: location_of_certificate_wallet_password_file`	Location of the file containing the encrypted wallet password. You must update this password by using the Directory Integration and Provisioning Assistant. **See Also:** The chapter on SSL and the directory in *Oracle Internet Directory Administrator's Guide* *Oracle Identity Management User Reference*

As an example, an `odi.properties` file can look like this:

```
RegWalletFile:  /private/myhost/orahome/ldap/odi/conf
CertWalletFile:  /private/myhost/orahome/ldap/dipwallet
CertWalletPwdFile: /private/myhost/orahome/ldap/
```

In the preceding example, the file locations are absolute path names. In this example, the wallet file `ewallet.p12` is located in the directory `/private/myhost/orahome/ldap/dipwallet`.

Starting, Stopping, and Restarting the Oracle Directory Integration and Provisioning Server

This section tells you how to start, stop, and restart the Oracle directory integration and provisioning server. It contains these topics:

- Starting the Oracle Directory Integration and Provisioning Server
- Stopping the Oracle Directory Integration and Provisioning Server
- Restarting the Oracle Directory Integration and Provisioning Server

> **Note:** When the Oracle directory integration and provisioning server is invoked in the default mode, it supports only the Oracle Provisioning Service, and not the Oracle Directory Synchronization Service.

Starting the Oracle Directory Integration and Provisioning Server

Oracle Directory Integration and Provisioning can be installed as a component of Oracle Internet Directory or as a standalone installation. How you start the Oracle directory integration and provisioning server depends on whether you install Oracle

Directory Integration and Provisioning as a component of Oracle Internet Directory as a standalone installation.

To start Oracle Directory Integration and Provisioning as a component of Oracle Internet Directory, you use the Oracle Internet Directory Monitor (`oidmon`) and the Oracle Internet Directory Control Utility (`oidctl`). You can start both utilities at the same time by using the Oracle Process Manager and Notification Server Control Utility (`opmnctl`). When you install Oracle Directory Integration and Provisioning as a component of Oracle Internet Directory, an instance of the Oracle directory integration and provisioning server is started that only processes provisioning requests. To start an additional instance of Oracle directory integration and provisioning server that performs synchronization, you must use the Oracle Internet Directory Control Utility (`oidctl`). The `oidmon`, `oidctl`, and `opmnctl` utilities are documented in the Oracle Identity Management server administration tools chapter of the *Oracle Identity Management User Reference*.

To start a standalone installation of Oracle Directory Integration and Provisioning, use the Oracle Directory Integration Server Control Tool (`odisrv`), which is also documented in the Oracle Identity Management server administration tools chapter of the *Oracle Identity Management User Reference*. In a standalone installation of Oracle Directory Integration and Provisioning, the Oracle directory integration and provisioning server instance starts by default if no other Oracle directory integration and provisioning server instance is running within the same Oracle Application Server infrastructure.

WARNING: If you attempt to manually stop then start the server within 30 seconds, the old server instance may not shut down before the new instance starts. This is because the Oracle Directory Integration and Provisioning server determines whether to shut down by polling the registration entry stored under `cn=odisrv,cn=subregistrysubentry` every 30 seconds. For this reason, be sure to wait for 30 seconds before restarting the server.

Stopping the Oracle Directory Integration and Provisioning Server

How you stop the directory integration and provisioning server depends on the utility you used to start it. If you started the server with either the `oidctl` or the `opmnctl` utilities, then you must the `oidctl` utility to stop it. If you used the `odisrv` utility to start the server, you must use the `stopodiserver.sh` command to stop it. You can also use `opmnctl` command to stop all running Oracle Internet Directory instances on a particular node, including directory servers, directory replication server, and directory integration and provisioning server. The `oidctl`, and `opmnctl`, `odisrv`, and `stopodiserver.sh` utilities are documented in the Oracle Identity Management server administration tools chapter of the *Oracle Identity Management User Reference*.

Restarting the Oracle Directory Integration and Provisioning Server

To restart the Oracle directory integration and provisioning server, first stop the server using the procedures described in "Stopping the Oracle Directory Integration and Provisioning Server" on page 4-9, wait 30 seconds, then start the server again using the procedures described in "Starting the Oracle Directory Integration and Provisioning Server" on page 4-8. You need to wait 30 seconds because the Oracle Directory Integration and Provisioning server determines whether to shut down by polling the registration entry stored under `cn=odisrv,cn=subregistrysubentry` at 30

second intervals. If you start the server before the next polling interval, the first instance of the server will not be stopped, resulting in two running instances.

Starting and Stopping the Oracle Directory Integration and Provisioning Server in a High Availability Scenario

The Oracle directory integration and provisioning server can, with certain restrictions, execute in various high availability scenarios. This section discusses the Oracle directory integration and provisioning server as it operates in a Real Application Clusters environment and in an Oracle Application Server Cold Failover Cluster (Infrastructure). It contains these topics

- The Oracle Directory Integration and Provisioning Server in a Real Application Clusters Environment
- The Oracle Directory Integration and Provisioning Server in an Oracle Application Server Cold Failover Cluster (Infrastructure)

In either type of high availability environment, there are two common scenarios for configuring Oracle Directory Integration and Provisioning. They are:

- Collocated–The Oracle directory integration and provisioning server is located within the cluster on the same node as Oracle Internet Directory.
- Outside the cluster–The Oracle directory integration and provisioning server is installed on a separate node, outside the cluster.

The Oracle Directory Integration and Provisioning Server in a Real Application Clusters Environment

The Oracle Internet Directory infrastructure is configured to work in a Real Application Clusters mode. In Real Application Clusters, the Oracle directory integration and provisioning server can execute against any directory node.

A particular configuration set can be executed by only one instance of the Oracle directory integration and provisioning server. For this reason, during the default installation only one server instance—namely, instance 1—is started on the Real Application Clusters master node. This server instance executes configuration set 0. Although it is started only on the master node, the server is nevertheless registered on all the nodes.

If the master node fails, then the Oracle directory integration and provisioning server instance is started by the OID Monitor on a secondary node. If there are multiple secondary nodes, then the server is started by the first OID Monitor to recognize the master node failure.

When it starts the server, the OID Monitor uses the same instance number and configuration set that was used on the master node. This is a transparent to the end user, and, once it is done, the Oracle directory integration and provisioning server on the secondary node behaves as if it is the primary server. The server continues executing on the secondary node as long as that node is available.

Two separate instances of the Oracle directory integration and provisioning server running on two nodes cannot simultaneously execute the same configuration set. Although the OID Monitor does not check for this, the Oracle directory integration and provisioning server itself fails to start.

You can stop the Oracle directory integration and provisioning server at any time by using the OID Control utility. However, if you do this, then the server does not start

automatically on any other node. To start it on another node, do so manually by using the OID Control utility.

If you execute the command `opmnctl stopall`, and subsequently execute `opmnctl startall`, then the Oracle directory integration and provisioning server starts.

In summary, unless an OID Control command stops the Oracle directory integration and provisioning server, the OID Monitor always ensures that the server is running.

Collocated Configurations

In a collocated configuration, you can start Oracle Directory Integration and Provisioning from any node in the cluster. Once the Oracle directory integration and provisioning server is started on the first node, you do not need to start it on any other node. On failure of the Oracle Directory Integration and Provisioning node, another node in OracleAS Cluster (Identity Management) will detect the failure and start the Oracle directory integration and provisioning server. No additional OID Control command is required to register the Oracle directory integration and provisioning server.

In most cases, the Oracle Directory Integration and Provisioning server communicates with only the single, default instance of the Oracle directory server. It is possible, however, to have manually configured the Oracle directory integration and provisioning server to communicate with a second instance of the Oracle directory server. If the second instance of the Oracle directory server is not configured on the other nodes, then on failover, the surviving node will start both Oracle Directory Integration and Provisioning and a second instance of the Oracle directory server.

In a collocated configuration, node failure is handled as follows: the OID Monitor on a surviving node keeps polling all other nodes every 10 seconds. When a node detects that one node is not responding, the OID Monitor on the surviving node starts the Oracle directory integration and provisioning server and possibly the LDAP server (if it is not on the default node).

Outside-the-Cluster Configurations

In an outside-the-cluster configuration, the Oracle directory integration and provisioning server node does not have failover capability. In this configuration, you can configure Oracle Directory Integration and Provisioning to connect to the Oracle Internet DirectoryLDAP server using a load balancer or virtual server in front of the multiple Oracle Internet Directory nodes.

The Oracle Directory Integration and Provisioning Server in an Oracle Application Server Cold Failover Cluster (Infrastructure)

In this configuration, you should start the Oracle directory integration and provisioning server with a virtual hostname. This is the default configuration on installation.

If the active node fails, then the OID Monitor on a standby node starts the Oracle directory integration and provisioning server instance on the standby node. When it does this, it uses the same instance number and configuration set as previously used on the active node. This is a transparent to the end user. The server continues executing on the active node as long as the node is available. In an Oracle Application Server Cold Failover Cluster (Infrastructure), the server is registered once for both the active and standby nodes because the virtual host names are the same for both.

You can stop the Oracle directory integration and provisioning server at any time by using the OID Control utility. However, if you do this, then the server does not start

again on this node. Moreover, if this node fails over, then the OID Monitor on the standby node does not start the Oracle directory integration and provisioning server. To start the server, you must use the OID Control utility.

If you execute the command `opmnctl stopall`, and subsequently execute `opmnctl startall`, then the Oracle directory integration and provisioning server starts.

In summary, unless an OID Control command stops the Oracle directory integration and provisioning server, OID Monitor always ensures that the server is running.

> **See Also:** The chapters on Oracle Application Server Cold Failover Cluster (Infrastructure) in *Oracle Application Server High Availability Guide*

Collocated Configurations

In a collocated configuration, start the Oracle Directory Integration and Provisioning server using this command line:

```
oidctl connect=connStr host=virtualHost server=odisrv instance=1 \
      flags="host=virtualHost port=OIDPORT" start
```

Outside-the-Cluster Configurations

In an outside-the-cluster configuration, use this command to start the Oracle Directory Integration and Provisioning server:

```
oidctl connect=connStr server=odisrv instance=1 \
      flags="host=OIDvirtualHost port=OIDPORT" start
```

> **Note:** There are two `host` parameters in the command-line examples for the collocated and outside-the-cluster configurations:
>
> - The `host` parameter outside the flags specifies the node where the OID Control utility runs and originates requests to the OID Monitor
> - The `host` parameter inside the flags specifies the LDAP server that the Oracle Directory Integration and Provisioning and replication servers should connect to. This parameter is valid only for those servers.

Setting the Debug Level for the Oracle Directory Integration and Provisioning Server

You set the debug level by specifying a value for the `orclodipdebuglevel` attribute in the profile. The value you assign to the `orclodipdebuglevel` attribute enables you to separately control the trace logging levels for the directory integration and provisioning server and that of each connector.

For server execution, tracing is stored in the $*ORACLE_HOME*/ldap/log/odisrv_*xx*.log file, where xx is the number of the started instance. For connectors, tracing is stored in the *$ORACLE_HOME*/ldap/odi/log/*profilename*.trc.

> **See Also:** Appendix C, "Troubleshooting Oracle Directory Integration and Provisioning" for more information on how trace and log files

Table 4–3 lists the server debugging levels you can assign to the `orclodipdebuglevel` attribute. If you specify a nonzero debug level, then each trace statement in the server log file includes these trace-statement types:

- `Main`–Messages from the controller thread
- `Scheduler`–Messages from the scheduler thread

Table 4–3 Server Debugging Levels

Debug Event Type	Numeric Value
Starting and stopping of threads	1
Refreshing of profiles	2
Initialization, execution, and end details of connectors	4
Details during connector execution	8
Change record of the connector	16
Mapping details of the connector	32
Execution time details of the connector	64

> **See Also:** Chapter 7, "Administration of Directory Synchronization" for instructions on selectively debugging the threads

If you do not set a value for the debug flag, then the default level is `0` (zero), and none of the debug events in Table 4–3 on page 4-13 are logged. However, errors and exceptions are always logged.

You can set the debugging levels for each connector in the profile itself. Table 4–4 lists the connector debugging levels you can assign to the `orclodipdebuglevel` attribute.

Table 4–4 Connector Debugging Levels

Debug Event Type	Numeric Value
Initializing and terminating	1
Searching within the connection	2
Processing of entries after searching	4
Creation of change records	8
Processing details of change records	16
Mapping details	32

> **See Also:** The `oidprovtool` section in the Oracle Directory Integration and Provisioning tools chapter of the *Oracle Identity Management User Reference* for information about the debug attribute for a synchronization profile

Managing Oracle Directory Integration and Provisioning in a Replicated Environment

For provisioning and synchronization, the replicated directory is different from the master directory. Any profiles created in the original directory need to be re-created in the new directory, and all configurations must be performed as in the original directory. To

Finding the Log Files

Execution details and debugging information are in the log file located in the $*ORACLE_HOME*/ldap/log/odisrv*Instance_number*.log directory.

For example, if the server was started as server instance number 3, then the log file would have this path name: $*ORACLE_HOME*/ldap/log/odisrv03.log.

Any other exceptions in the server are in the file odisrv_jvm_*xxxx*.log where *xxxx* is the identifier of the process running the directory integration and provisioning server in that table.

All the profile-specific debug events are stored in the profile-specific trace file in $*ORACLE_HOME*/ldap/odi/log/*profile_name*.trc.

Manually Registering the Oracle Directory Integration and Provisioning Server

The Oracle directory integration and provisioning server is registered with Oracle Internet Directory during installation of Directory Integration and Provisioning. This registration creates a footprint in the directory indicating the specified host as the one authorized to run Directory Integration and Provisioning.

There may be times when you need to perform this registration manually on the client side, as, for example, if there is a failure during installation. You can do this by using either the Oracle directory integration and provisioning server registration tool (`odisrvreg`) or Oracle Enterprise Manager 10*g* Application Server Control Console.

You must separately register each directory integration and provisioning server on each host by running `odisrvreg` on that host. To run this tool, you need privileges to administer a directory server.

See Also:

- The `oidsrvreg` section in the Oracle Directory Integration and Provisioning tools chapter of the *Oracle Identity Management User Reference* for instructions on using `oidsrvreg`
- "Troubleshooting Synchronization" on page C-19

Manually Registering the Oracle Directory Integration and Provisioning Server by Using Oracle Enterprise Manager 10*g* Application Server Control Console

You can use Oracle Enterprise Manager 10*g* Application Server Control Console to configure Directory Integration and Provisioning in an Oracle Identity Management infrastructure. When you do this, Application Server Control Console registers the Oracle directory integration and provisioning server on that infrastructure.

1. On the main Application Server Control Console page, select the name of the Oracle Application Server instance you want to manage in the **Standalone**

Instances section. The Oracle Application Server home page opens for the selected instance.

2. Select the **Configure Components** button, located just above the System Components table. The Select Component page appears.

   Note: The Configure Component button is available only if you have installed but not configured any Oracle Application Server components.

3. Select **Oracle Directory Integration and Provisioning**, then select **Continue**. The Login screen appears.
4. Enter the user name and password of the directory super user. The default user name is `cn=orcladmin`.
5. Select **Finish** to complete the registration.

Part III

Synchronization in Oracle Identity Management Integration

This part discusses the concepts and components involved in synchronization between Oracle Identity Management and other identity management systems. It also discusses things you should consider when deciding how to deploy synchronization. It explains how to configure and run synchronization.

5

Oracle Directory Synchronization Service

This chapter discusses the synchronization profiles and connectors that link Oracle Internet Directory and connected directories. It contains these topics:

- Components Involved in Oracle Directory Synchronization
- How Synchronization Works

> **Tip:** Chapter 1, "Introduction to Oracle Identity Management Integration" for a conceptual discussion of Oracle Directory Integration and Provisioning

Components Involved in Oracle Directory Synchronization

This section contains these topics:

- Connectors for Directory Synchronization
- Directory Synchronization Profiles

Connectors for Directory Synchronization

To synchronize between Oracle Internet Directory and a connected directory, Oracle Directory Integration and Provisioning relies on a prepackaged connectivity solution called a connector. Minimally, this connector consists of a **directory integration profile** containing all the configuration information required for synchronization.

Using Connectors with Supported Interfaces

When synchronizing between Oracle Internet Directory and a connected directory, Directory Integration and Provisioning uses one of these interfaces: DB, LDAP, tagged, or LDIF. If the connected directory uses one of these interfaces, then the connector requires only a directory integration profile for synchronization to occur. For example, the SunONE connector provided with Oracle Internet Directory uses the LDAP interface to read the changes from the SunONE Directory Server. The changes are in the format specific to SunONE Directory Server and can be determined by doing an ldapsearch in the SunONE Directory Server.

Using Connectors Without Supported Interfaces

If a connected directory cannot use one of the interfaces supported by Directory Integration and Provisioning, then, in addition to the directory integration profile, it requires an agent. The agent transforms the data from one of the formats supported by Directory Integration and Provisioning into one supported by the connected directory. An example is the Oracle Human Resources connector. It has both a prepackaged integration profile and an Oracle Human Resources agent. To communicate with

Oracle Internet Directory, the agent uses the tagged file format supported by Directory Integration and Provisioning. To communicate with the Oracle Human Resources system, it uses SQL (through an OCI interface).

Directory Synchronization Profiles

A directory integration profile for synchronization, called a **directory synchronization profile**, contains all the configuration information required for synchronization including:

- Direction of Synchronization

  Some connected directories only receive data from Oracle Internet Directory—that is, they participate in export operations only. Others only supply data to Oracle Internet Directory—that is, they participate in import operations only. Still others participate in both import and export operations.

  A separate profile is used for each direction—that is, one profile for information coming into Oracle Internet Directory, and another for information going from Oracle Internet Directory to connected directories.

- Type of Interface

  Some connected directories can receive data in any of the interfaces built into Directory Integration and Provisioning. These interfaces include LDAP, tagged, DB (for read-only), and LDIF. For these connected directories, the Oracle Directory Synchronization Service performs the synchronization itself directly, using the information stored in the profile.

- Mapping Rules and Formats

  In a directory synchronization environment, a typical set of entries from one domain can be moved to another domain. Similarly, a set of attributes can be mapped to another set of attributes.

  Mapping rules govern the conversion of attributes between a connected directory and Oracle Internet Directory. Each connector stores a set of these rules in the `orclodipAttributeMappingRules` attribute of its synchronization profile. The Oracle directory integration and provisioning server uses these rules to map attributes as needed when exporting from the directory and interpreting data imported from a connected directory or file. When the Oracle directory integration and provisioning server imports changes into Oracle Internet Directory, it converts the connected directory's change record into an LDAP change record following the mapping rules. Similarly, during export, the connector translates Oracle Internet Directory changes to the format understood by the connected directory.

- Connection details of the connected directory

  These details include such information about the connected directory as host, port, mode of connection—that is, either SSL or non-SSL—and the connected directory credentials.

- Other Information

  Although the synchronization profile stores most of the information needed by a connector to synchronize Oracle Internet Directory with connected directories, some connectors may need more. This is because some operations might require additional configuration information at runtime.

  You can store such additional connector configuration information wherever and however you want. However, Directory Integration and Provisioning enables you to store it in the synchronization profile as an attribute called

`orclODIPAgentConfigInfo`. Its use is optional—that is, if a connector does not require such information, then simply leave this attribute empty.

This configuration information can pertain to the connector, the connected directory, or both. Oracle Internet Directory and Oracle directory integration and provisioning server do not modify this information. When the connector is invoked, the Oracle directory integration and provisioning server simply provides it with the information in this attribute as a temporary file.

> **See Also:** The attribute reference chapter of the *Oracle Identity Management User Reference* for a list and descriptions of the attributes in a directory integration profile

How Synchronization Works

Depending on where the changes are made, synchronization can occur:

- From a connected directory to Oracle Internet Directory
- From Oracle Internet Directory to a connected directory
- In both directions

Regardless of the direction in which the data flows, it is assumed that:

- During synchronization, incremental changes made on one directory are propagated to the other
- Once synchronization is complete, the information maintained on both directories is the same

This section contains these topics:

Synchronizing from Oracle Internet Directory to a Connected Directory

Oracle Internet Directory maintains a change log in which it stores incremental changes made to directory objects. It stores these changes sequentially based on the change log number.

Synchronization from Oracle Internet Directory to a connected directory makes use of this change log. Consequently, when running the Oracle directory integration and provisioning server, you must start Oracle Internet Directory with the default setting in which change logging is enabled. If change logging is disabled, you can enable it by using the `-l` flag in the OID Control Utility (`oidctl`) as described in the *Oracle Identity Management User Reference*.

Each time the Oracle Directory Synchronization Service processes a synchronization profile, it:

1. Retrieves the latest change log number up to which all changes have been applied
2. Checks each change log entry more recent than that number
3. Selects changes to be synchronized with the connected directory by using the filtering rules in the profile

4. Applies the mapping rules to the entry and makes the corresponding changes in the connected directory

The appropriate entries or attributes are then updated in that connected directory. If the connected directory does not use DB, LDAP, tagged, or LDIF formats directly, then the agent identified in its profile is invoked. The number of the last change successfully used is then stored in the profile.

Periodically, Oracle Internet Directory purges the change log after all profiles have used what they need, and identifies where subsequent synchronization should begin.

Synchronizing from a Connected Directory to Oracle Internet Directory

When a connected directory uses DB, LDAP, tagged, or LDIF formats directly, changes to its entries or attributes can be automatically synchronized by the Oracle Directory Synchronization Service. Otherwise, the connector has an agent in its synchronization profile, which writes the changes to a file in the LDIF or tagged format. The Oracle Directory Synchronization Service then uses this file of connected directory data to update Oracle Internet Directory.

Synchronizing with Directories with Interfaces Not Supported by Oracle Internet Directory

Some connected directories cannot receive data by using any of the interfaces supported by Oracle Internet Directory. Profiles for this type of directory contain an attribute identifying a separate program for synchronization, called an agent. The agent translates between the connected directory's unique format and a DB, LDAP, tagged, or LDIF file containing the synchronization data. The agent, as identified in the profile, is invoked by the Oracle Directory Synchronization Service.

When exporting data from Oracle Internet Directory to this type of connected directory, the Oracle Directory Synchronization Service creates the necessary file in the tagged or LDIF format. The agent then reads that file, translates it into the correct format for the receiving connected directory, and stores the data in that directory.

When importing data from this type of connected directory to Oracle Internet Directory, the agent creates the necessary tagged or LDIF format file. The Oracle Directory Synchronization Service then uses this file data to update the Oracle Internet Directory.

6

Configuration of Directory Synchronization Profiles

This chapter explains how to register connectors with Oracle Directory Integration and Provisioning and how to format the mapping rule attribute. It contains these topics:

- Registration of Connectors into Oracle Directory Integration and Provisioning
- Sample Synchronization Profiles
- Configuring Connection Details
- Additional Configuration Information
- Configuring Mapping Rules
- Applying Matching Filters
- Location and Naming of Files

> **See Also:** The attribute reference chapter of the *Oracle Identity Management User Reference* for a list and descriptions of the attributes in synchronization profiles

Registration of Connectors into Oracle Directory Integration and Provisioning

Before deploying a connector, you register it in Oracle Internet Directory. This registration involves creating a directory synchronization profile, which is stored as an entry in the directory.

To create a directory synchronization profile, use one of the following tools:

- The Oracle Directory Integration and Provisioning Server Administration tool
- Directory Integration and Provisioning Assistant

> **See Also:**
>
> - Chapter 3, "Oracle Directory Integration and Provisioning Administration Tools"
> - "Directory Synchronization Profiles" on page 5-2

Attributes in a synchronization profile entry belong to the object class `orclodiProfile`. The only exception is the

orclodiplastappliedchangenumber attribute, which belongs to the object class orclchangesubscriber.

The Object Identifier prefix 2.16.840.1.113894.7 is assigned to platform-related classes and attributes.

The various synchronization profile entries in the directory are created under the container cn=subscriber profile,cn=changelog subscriber,cn=oracle internet directory. For example, a connector called OracleHRAgent is stored in the directory as orclodipagentname=OracleHRAgent,cn=subscriber profile,cn=changelog subscriber,cn=oracle internet directory.

Sample Synchronization Profiles

When you install Oracle Directory Integration and Provisioning, sample import and export synchronization profiles are automatically created for:

- Microsoft Active Directory
- SunONE (iPlanet) Directory Server
- LDIF files
- Tagged files

The property and mapping files used to create the sample profiles are available in the $*ORACLE_HOME*/ldap/odi/samples and $*ORACLE_HOME*/ldap/odi/conf directory directories.

Configuring Connection Details

Some of the most important pieces of a directory synchronization profile include the connection details you assign to the properties listed in Table 6–1:

Table 6–1 Connection detail properties

Property	Description
odip.profile.condirurl	The URL of the connected directory: ■ To connect to an LDAP directory, use the form *host*:*port* ■ To connect in SSL mode, use the form *host*:*port*:1. ■ To connect to a database, use the form *host*:*port*:*sid*
odip.profile.condiraccount	The DN or account name used to connect to the third-party directory
odip.profile.condirpassword	The password used to connect to the third-party directory

Notes:

- The account information you specify must have sufficient privileges in the directory to which you are connecting.
- The account name and password properties are not required if you are using the LDIF or tagged data formats.

Additional Configuration Information

The Additional Config Info (`orclodipAgentConfigInfo`) attribute in a synchronization profile stores any additional configuration information needed by a connector to synchronize Oracle Internet Directory with a connected directory. Although not required, you can use the following two parameters with the Additional Config Info attribute to significantly improve synchronization efficiency:

- The SearchDeltaSize Parameter
- The SkipErrorToSyncNextChange Parameter

You cannot use the Oracle Directory Integration and Provisioning Server Administration tool or Oracle Directory Manager to modify the Additional Config Info attribute. Instead, you must use the Directory Integration and Provisioning Assistant.

See Also: *Oracle Identity Management User Reference*

The SearchDeltaSize Parameter

The `SearchDeltaSize` parameter determines how many incremental changes are processed during each iteration in a synchronization cycle. By default, the `SearchDeltaSize` parameter is assigned a value of 500. In some cases, you will experience better synchronization efficiency if you assign a higher value to this parameter. However, be sure that the value you specify does not exceed the LDAP search limit of the connected directory server. Otherwise, you may receive an error during synchronization and some changes may not be processed.

WARNING: Be sure to thoroughly analyze and test your deployment when modifying the `SearchDeltaSize` parameter, especially if you assign a value over 2,000.

The SkipErrorToSyncNextChange Parameter

The `SkipErrorToSyncNextChange` parameter determines how the Oracle directory integration and provisioning server handles an error when processing a change during synchronization. By default, the `SkipErrorToSyncNextChange` parameter is assigned a value of `false`, which means that the Oracle directory integration and provisioning server will continue processing a change until the error is resolved. If you assign a value of `true` to the `SkipErrorToSyncNextChange` parameter, the Oracle directory integration and provisioning server will skip any changes that cause an error. Any failures are recorded in the $*ORACLE_HOME*/ldap/odi/log/*profilename*.aud audit log file. If you do assign a value of `true` to the `SkipErrorToSyncNextChange` parameter, be sure to periodically review the audit log for failures.

See Also: "Troubleshooting Synchronization" on page C-19

Configuring Mapping Rules

This section discusses how to configure mapping rules. It contains these topics:

- Distinguished Name Mapping
- Attribute-Level Mapping
- How to Construct a New Mapping File

The mapping rules attribute enables you to specify how to convert entries from one directory to another. There are two types of mapping rules: domain rules and attribute rules. You can specify distinguished name mapping and attribute-level mapping. This attribute is assumed to be in the format of a file as described in this section.

Mapping rules are organized in a fixed tabular format, and you must follow that format carefully. Each set of mapping rules appears between a line containing only the word `DomainRules` and a line containing only the characters `###`. The fields within each rule are delimited by a colon (:).

```
DomainRules
srcDomainName1: [dstDomainName1]: [DomainMappingRule1]
srcDomainName2: [dstDomainName2]: [DomainMappingRule2]
AttributeRules
srcAttrName1:[ReqAttrSeq]:[SrcAttrType]:[SrcObjectClass]:[dstAttrName1]:
[DstAttrType]:[DstObjectClass]:[AttrMappingRule1]
srcAttrName1,srcAttrName2:[ReqAttrSeq]:[SrcAttrType]:[SrcObjectClass]:
[dstAttrName2]:[DstAttrType]:[DstObjectClass]:[AttrMappingRule2]
###
```

where the expansion of each *srcAttrName1* and *srcAttrName2* would be a single, unwrapped long line.

Distinguished Name Mapping

This section specifies how entries are mapped between Oracle Internet Directory and a connected directory. If the mapping is between Oracle Internet Directory and another LDAP directory, then you can create multiple mapping rules, as explained in "Configuring Mapping Rules" on page 6-3. The domain rule specifications appear after a line containing only the keyword `DomainRules`. Each domain rule is represented with the components, separated by colons, that are described in Table 6–2.

Table 6–2 DomainRule Components

Component Name	Description
`SrcDomainName`	Name of the domain or container of interest. Specify NONLDAP for sources other than LDAP and LDIF.
`DstDomainName`	Name of the domain of interest in the destination. Specify this component if the container for the entries in the destination directory is different from that in the source directory. If the value assigned to `SrcDomainName` is an LDAP or LDIF domain, then this field assumes the same value. However, if the value assigned to `SrcDomainName` is not an LDAP or LDIF domain, you must specify the container where entries should be created. If not specified, this field assumes the value of `SrcDomainName` under valid conditions. For destinations other than LDAP and LDIF, specify `NONLDAP`. Because "import" and "export" always refer to Oracle Internet Directory, a combination of `NONLDAP:NONLDAP` is not allowed.

Table 6–2 (Cont.) DomainRule Components

Component Name	Description
`DomainMappingRule`	This rule is used to construct the destination DN from the source domain name, from the attribute given in `AttributeRules`, or both. This field is typically of the form `cn=%,l=%,o=oracle,dc=com`. Such specifications are used to put entries under different domains or containers in the directory. In case of non-LDAP sources, this rule specifies how to form the target DN so it can add entries to the directory. This field is meaningful only when importing to Oracle Internet Directory, or when exporting to an LDIF file or another external LDAP-compliant directory. Specify this component if any part of an entry's DN in the destination directory is different from that in the source directory entry. This component is optional in LDAP-to-LDIF, LDAP-to-LDAP, or LDIF-to-LDAP. If it is not specified, then the source domain and destination domain names are considered to be the same.

See Also: The mapping file examples at the end of this chapter

Attribute-Level Mapping

The attribute rule specifications appear after a line containing only the keyword `AttributeRules`. Attribute rules specify how property values for an entry are related between two LDAP directories. For example, the `cn` attribute of a user object in one directory can be mapped to the `givenname` object in another directory. Similarly, the `cn` attribute of a group object in one directory can be mapped to the `displayname` attribute in another directory. Each attribute rule is represented with the components, separated by colons, and described in Table 6–3. The attribute rule specifications end with a line containing only the characters `###`.

Table 6–3 Components in Attribute Rules

Component Name	Description
`SrcAttrName`	For LDAP-compliant directory repositories, this parameter refers to the name of the attribute to be translated. For Oracle Database repositories, it refers to the ColumnName in the table specified by the `SrcClassName`. For other repositories this parameter can be appropriately interpreted.
`ReqAttrSeq`	Indicator of whether the source attribute must always be passed to the destination. When entries are synchronized between Oracle Internet Directory and the connected directory, some attributes need to be used as synchronization keys. This field indicates whether the specified attribute is being used as a key. If so, regardless of whether the attribute has changed or not, the value of the attribute is always extracted from the source. A nonzero integer value should be placed in this field if the attribute needs to be always passed on to the other end.
`SrcAttrType`	This parameter refers to the attribute type—for example, integer, string, binary—that validates the mapping rules.

Table 6–3 (Cont.) Components in Attribute Rules

Component Name	Description
SrcObjectClass	If the source of the shared attribute is an LDAP-compliant directory, then this parameter names the object class to which the attribute belongs. If the source of the shared attribute is an Oracle Database repository, then this parameter refers to the table name and is mandatory. For other repositories, this parameter may be ignored.
DstAttrName	Optional attribute. If it is not specified, then the SrcAttrName is assumed. For LDAP-compliant directories, this parameter refers to the name of the attribute at the destination. For Oracle Database repositories, it refers to the ColumnName in the table specified by the SrcClassName. For other repositories, this parameter can be appropriately interpreted.
DstAttrType	This parameter refers to the attribute type—for example, integer, string, binary. Note that it is up to you, the administrator, to ensure the compatibility of the source and destination attribute types. Directory Integration and Provisioning does not ensure this compatibility.
DstObjectClass	For LDAP-compliant directories, this parameter refers to the object class to which the attribute belongs, and is optional. For Oracle Database repositories, it refers to the table name, and is mandatory. For other repositories this parameter may be ignored.
AttrMapping Rule	Optional arithmetic expression with these operators: +, |, and these functions: toUpper (string),toLower(String), trunc (string,char). If nothing is specified, then the source attribute value is copied as the value of the destination attribute. Literals can be specified with single quotes ('') or with double quotes ("").

In a newly created synchronization profile, mapping rules are empty. To enter mapping rules, edit a file that strictly follows the correct format.

> **Note:** When attributes and object classes are defined in the mapping file, it is assumed that source directories contain the respective attributes and object classes defined in the schema.
>
> If a parent container is selected for synchronization, then all its children that match the mapping rules are likewise synchronized. Child containers cannot be selectively ignored for synchronization.

How to Construct a New Mapping File

To create a new mapping file, follow these steps:

1. Identify the container(s) of interest for synchronization in the source directory.
2. Identify the destination container or containers to which the objects in the source containers should be mapped to. Be sure that the specified container already exists in the directory.

3. Determine the rule to create a DN of the entry to be created in the destination directory. In LDAP-to-LDAP, mapping is normally one-to-one. In non-LDAP-to-LDAP, a domain, DN construct rule is required. For instance in the case of synchronizing from a tagged file or Human Resources agent, the mapping rule may be of the form `uid=%,dc=mycompany,dc=com`. In this case, the `uid` attribute must be present in all the changes to be applied from Oracle Human Resources. The `uid` attribute must be specified as a required attribute, as specified in step 6.
4. Identify the objects that you want to synchronize between directories—that is, the relevant object classes in the source and destination directories. In general, objects that get synchronized between directories include users, groups, organizational units, organizations, and other resources. Identify the actual object classes used in the directories to identify these objects.
5. Identify the properties of the various objects that you want to synchronize between directories—that is, the attributes in the LDAP context. All the attributes of an object need not be synchronized. The properties of users that you might want to synchronize are `cn`, `sn`, `uid`, `mail`.
6. Define the mapping rules. Each mapping rule has this format:

   ```
   <srcAttrName>:<ReqdFlag>:<srcAttrType>:<SrcObjectClass>:
   <dstAttrName>:<dstAttrType>:<dstObjectClass>: <Mapping Rule>
   ```

   While defining the mapping rule, ensure the following:

   - Every required attribute has a sequence number. For example, if in step 3 the `uid` attribute is identified as required, then assign a value of `1` in place of `<ReqdFlag>`.
   - Every relevant object class has a schema definition on the destination directory.
   - Every mandatory attribute in a destination object class has a value assigned from the source. This holds good even for standard object classes also, as the different LDAP implementations may not be completely standards-compliant.

   It is not necessary to assign all attributes belonging to a source object class to a single destination object class. Different attributes of a source object class can be assigned to different attributes belonging to different destination object classes.

   If an attribute has binary values, then specify it as `binary` in the `<attrtype>` field.

Mapping rules are flexible: They can include both one-to-many and many-to-one mappings.

- One-to-many

  One attribute in a connected directory can map to many attributes in Oracle Internet Directory. For example, suppose an attribute in the connected directory is `Address:123 Main Street/MyTown, MyState 12345`. You can map this attribute in Oracle Internet Directory to both the LDAP attribute `homeAddress` and the LDAP attribute `postalAddress`.

- Many-to-one

  Multiple attributes in a connected directory can map to one attribute in Oracle Internet Directory. For example, suppose that the Oracle Human Resources directory represents Anne Smith by using two attributes: `firstname=Anne` and `lastname=Smith`. You can map these two attributes to one attribute in Oracle

Internet Directory: `cn=Anne Smith`. *However, in bidirectional synchronization, you cannot then map in reverse. For example, you cannot map cn=Anne Smith to many attributes.*

> **See Also:** The mapping file examples at the end of this chapter

Supported Attribute Mapping Rules and Examples

The attribute mapping rules supported are:

- `Concatenation (+)`: Used to concatenate two string attributes

  The mapping rule looks like:

  ```
  Firstname,lastname:  :  :  : givenname:  : inetorgperson: firstname+lastname
  ```

  For example, if the `Firstname` is `John` and `LastName` is `Doe` in the source, then this rule results in the `givenname` attribute in the destination with the value `JohnDoe`.

- OR operator (`|`): Used to assign one of the values of the two string attributes to the destination

  The mapping rule looks like this:

  ```
  Fistname,lastname : : : :givenname: :inetorgperson: firstname | lastname
  ```

  In this example, `givenname` is assigned the value of `firstname` if it exists. If the `firstname` attribute does not exist, then `givenname` is assigned the value of `lastname`. If both the values are empty, then no value is assigned.

- `bin2b64 ( )`: Used to store a binary value of the source directory as a base64 encoded value in the destination directory. Typical usage is as follows:

  ```
  objectguid: : : :binary: :orclobjectguid: orcladuser:bin2b64(objectguid)
  ```

  This is required when you need search on the value of (`objectguid`).

- `tolower()`: Convert the String attribute value to lowercase.

  ```
  firstname: : : :givenname: :inetorgperson: tolower(firstname)
  ```

- `toupper ()`: Convert the String attribute value to uppercase.

  ```
  firstname: : : :givenname: :inetorgperson: toupper(firstname)
  ```

- `trunc(str,char)`: Truncate the string beginning from the first occurrence of the specified char

  ```
  mail : : : : uid : : inetorgperson : trunc(mail,'@')
  ```

  For example, if `mail` is `John.Doe@acme.com` in the source, then this rule results in the `uid` attribute in the destination with the value "John.Doe"

- `truncl(str, char)`: Truncate the string up to and including the first occurrence of the specified char

  ```
  mail : : : : uid : : inetorgperson : truncl(mail,'@')
  ```

  For example, if mail is `John.Doe@acme.com` in the source, then this rule results in the `uid` attribute in the destination with the value `acme.com`.

- `trunc(str1, str2)`: Truncate the string beginning with the first occurrence of the specified string

  ```
  mail : : : : uid : : inetorgperson : trunc1(mail, "@")
  ```

- `dnconvert (str)`: Used for DN type attributes if domain mapping is used.

  This example assumes the following domain mapping rule:

  ```
  DomainRules
  cn=srcdomain:cn=dstdomain:
  ```

  For example:

  ```
  uniquemember : : : groupofuniquenames : uniquemember : :groupofuniquenames :
  dnconvert(uniquemember)
  ```

  In this example, if `uniquemember` in the source is `cn=testuser1,cn=srcdomain,` then `uniquemember` in the destination becomes `cn=test user1,cn=dstdomain`.

- Literals:

  ```
  Userpassword: : :person: userpassword: :person: 'welcome1'
  ```

Example: A Mapping File for a TAGGED-File Interface

Based on the preceding discussions, here is a sample mapping file for importing user entries from the Oracle Human Resources database tables by using the tagged-file interface. Note that the source is a non-LDAP directory. This sample file is supplied during installation, at $*ORACLE_HOME*/ldap/odi/conf/oraclehragent.map.master.

```
DomainRules
NONLDAP:dc=myCompany,dc=com:uid=%dc=myCompany,dc=com
AttributeRules
firstname: : : :cn: :person
email : : : :cn: :person: trunc(email,'@')
email : 1 : :uid: :person:trunc(email,'@')
firstname,lastname: : : :cn: :person: firstname+","+lastname
lastname,firstname: : : :cn: :person: lastname+","+firstname
firstname,lastname: : : :sn: :person: lastname | firstname
EmployeeNumber: : : :employeenumber: :inetOrgperson
EMail: : : :mail: :inetOrgperson
TelephoneNumber1: : : :telephonenumber: :person
TelephoneNumber2: : : :telephonenumber: :person
TelephoneNumber3: : : :telephonenumber: :person
Address1: : : :postaladdress: :person
state: : : :st: :locality
street1: : : :street: :locality
zip: : : :postalcode: :locality
town_or_city: : : :l: :locality
Title: : : :title: :organizationalperson
#Sex: : : :sex: :person
###
```

As described earlier, the mapping file consists of keywords and a set of domain and attribute mapping rule entries. The mapping file in this example contains the domain rule `NONLDAP:dc=myCompany,dc=com:cn=%,dc=myCompany,dc=com`.

- This rule implies that the source domain is NONLDAP—that is, there is no source domain.

- The destination domain (`:dc=myCompany,dc=com`) implies that all the directory entries this profile deals with are in the domain `dc=myCompany,dc=com`. Be sure that the domain exists before the start of synchronization.
- The domain mapping rule (`:uid=%,dc=myCompany,dc=com`) implies that the data from the source should refer to the entry in the directory with the DN that is constructed using this domain mapping rule. In this case, `uid` must be one of the destination attributes that should always have a non-null value. If any data corresponding to an entry to be synchronized has a null value, then the mapping engine assumes that the entry is invalid and proceeds to the next entry. To identify the entry correctly in the directory, it is also necessary that `uid` should be single-valued.
- In the case of the tagged file, the source entry does not have any object class to indicate the type of object it is synchronizing. Note that the `SrcObjectClass` field is empty.
- Every object whose destination is Oracle Internet Directory must have an object class. Specify an object class for every attribute.
- Note that `email` is specified as a required attribute in the sample mapping file. This is because the `uid` attribute is derived from the `email` attribute. Successful synchronization requires the `email` attribute to be specified in all changes specified in the tagged file as follows:

```
Email : 1 : : :uid : : person : trunc(email,'@')
```

- In some cases, the RDN of the DN needs to be constructed by using the name of a multivalued attribute. For example, to construct an entry with the DN of `cn=%,l=%,dc=myCompany,dc=com`, where `cn` is a multivalued attribute, the DomainMappingRule can be of this form: `rdn,l=%,dc=myCompany,dc=com` where `rdn` is one of the destination attributes having a non-null value. A typical mapping file supporting this could have the following form:

```
DomainRules
NONLDAP:dc=us,dc=myCompany,dc=com:rdn,l=%,dc=us,dc=myCompany,dc=com
AttributeRules
firstname: : :cn: :person
email : : : :cn: :person: trunc(email,'@')
email : 1: : :rdn: :person: 'cn='+trunc(email,'@')
firstname,lastname: : : :cn: :person: firstname+","+lastname
lastname,firstname: : : :cn: :person: lastname+","+firstname
firstname,lastname: : : :sn: :person: lastname | firstname
EmployeeNumber: : : :employeenumber: :inetOrgperson
EMail: : : :mail: :inetOrgperson
TelephoneNumber1: : : :telephonenumber: :person
TelephoneNumber2: : : :telephonenumber: :person
TelephoneNumber3: : : :telephonenumber: :person
Address1: : : :postaladdress: :person
Address1: : : :postaladdress: :person
Address1: : : :postaladdress: :person
state: : : :st: :locality
street1: : : :street: :locality
zip: : : :postalcode: :locality
town_or_city: 2 : : :1: :locality
Title: : : :title: :organizationalperson
#Sex: : : :sex: :person
###
```

Example: Mapping Files for an LDIF Interface

A set of sample integration profiles are created as part of installation by using the Directory Integration and Provisioning Assistant. The properties file used for creating the profile is located in the directory $*ORACLE_HOME*/ldap/odi/samples.

Sample Import Mapping File

```
DomainRules
dc=mycompany.oid,dc=com:dc=mycompany.iplanet,dc=com
AttributeRules
# Mapping rules to map the domains and containers
o: : :organization: o: :organization
ou: : :organizationalUnit: ou: : organizationalUnit
dc: : :domain:dc: :domain
# Mapping Rules to map users
uid : : :person: uid: :inetOrgperson
sn: : :person:sn: :person
cn: : :person:cn: :person
mail: :inetorgperson: mail: :inetorgperson
employeenumber: :organizationalPerson: employeenumber: :organizationalperson
c: : :country:c: :country
l: : :locality: l: :locality
telephonenumber: :organizationalPerson: telephonenumber: :organizationalperson
userpassword: : :person: userpassword: :person
uid: : :person: orcldefaultProfileGroup: :orclUserV2
# Mapping Rules to map groups
cn: : :groupofuniquenames:cn: :groupofuniquenames
member: : :groupofuniquenames:member: :orclgroup
uniquemember: : :groupofuniquenames:uniquemember: :orclgroup
owner: : :groupofuniquenames:owner: :orclgroup
# userpassword: :base64:userpassword: :binary:
```

Notice in the preceding example that both the source domain and destination domain are specified in the Domain Mapping rule section. In this example, the source and the destination domains are the same. However, you can specify a different destination domain, provided the container exists in the destination directory.

Also notice in the preceding example that the attribute rules are divided into two sections: user attribute mapping rules and group attribute mapping rules. Specifying the object class in a mapping rule helps to uniquely map a specific attribute of an object.

Updating Mapping Rules

You can customize mapping rules by adding new ones, modifying existing ones, or deleting some from the mapping rule set specified in the `orclodipAttributeMappingRules` attribute. In general, to perform any of these operations, you identify the file containing the mapping rules, or store the value of the attribute for a file by using an `ldapsearch` command as described in the *Oracle Internet Directory Administrator's Guide*.

You cannot edit the mapping rules in the Oracle Directory Integration and Provisioning Server Administration tool. Instead, mapping rules are stored in a file that you upload to the directory as a value of the attribute. To upload the mapping file, use the Directory Integration and Provisioning Assistant. Once you have created and uploaded the mapping file, you can maintain a copy of it in the $*ORACLE_HOME*/ldap/odi/conf directory, and upload it again after any future update.

```
dipassistant mp -profile profile name odip.profile.mapfile=map file
```

See Also: *Oracle Identity Management User Reference*

Adding an Entry to the Mapping Rules File

To add a new entry to the mapping rules file, edit this file and add a record to it. To do this:

1. Identify the connected directory attribute name and the object class that needs to be mapped to Oracle Internet Directory.
2. Identify the corresponding attribute name in Oracle Internet Directory and the object class to which it needs to be mapped.
3. Generate the mapping rule elements indicating the conversion that needs to be done on the attribute values.
4. Load the attribute mapping rule file to the synchronization profile.

   For instance, if the e-mail attribute of an entry in the source directory needs to be mapped to the unique identifier of the destination, then it can be:

   ```
   Email:  :  : inetorgperson: uid: : person:
   ```

Modifying an Entry in the Mapping Rules File

After you identify an entry to be modified in the mapping rules file, generate the mapping rule element for the desired conversion of attribute values.

Deleting an Entry from the Mapping Rules File

After you identify an entry to be deleted in the mapping rules file, you can either delete the entry from the file or comment it out by putting a hash mark (#) in front of it.

See Also:

- The Oracle Directory Integration and Provisioning tools chapter of the *Oracle Identity Management User Reference* for instructions on using the Directory Integration and Provisioning Assistant
- "Location and Naming of Files" on page 6-14 for the names of these files
- Oracle MetaLink Note: 261342.1—Understanding DIP Mapping Files available on Oracle MetaLink at `http://metalink.oracle.com/`

Note: To run shell script tools on the Windows operating system, you need one of the following UNIX emulation utilities:

- Cygwin 1.3.2.2-1 or later. Visit: `http://sources.redhat.com`
- MKS Toolkit 6.1. Visit: `http://www.datafocus.com/`

Applying Matching Filters

By default, a connector retrieves changes to all objects in the container configured for synchronization.However, you may be interested in synchronizing only certain types of changes, such as changes to just users and groups. While mapping rules allow you to specify how entries are converted from one directory to another, you can also filter objects that are synchronized between directories. Before changes from a connected directory are imported into Oracle Internet Directory, they can be filtered with the Connected Directory Matching Filter (`orclODIPConDirMatchingFilter`) attribute in the synchronization profile. Similarly, before changes are exported from Oracle Internet Directory to a connected directory, they can be filtered with the OID Matching Filter (`orclODIPOIDMatchingFilter`) attribute. For both attributes, you can specify a filter for connected directories that either obtain incremental changes through an LDAP search or that store changes in a change log, as described in the following sections:

- Filtering Changes with an LDAP Search
- Filtering Changes from a Change Log

You can use either the Oracle Directory Integration and Provisioning Server Administration tool or Directory Integration and Provisioning Assistant to update the matching filters.

See Also: Chapter 3, "Oracle Directory Integration and Provisioning Administration Tools"

Filtering Changes with an LDAP Search

For connected directories that obtain incremental changes through an LDAP search, such as Active Directory, use the following syntax to assign a value to the `searchfilter` attribute of either the Connected Directory Matching Filter (`orclODIPConDirMatchingFilter`) or the OID Matching Filter (`orclODIPOIDMatchingFilter`):

```
"searchfilter=LDAP_SEARCH_FILTER"
```

The following example creates an LDAP search filter that retrieves organizational units, groups, and users, but not computers:

```
"searchfilter=(|(objectclass=group)(objectclass=organizationalUnit)
(&(objectclass=user)(!(objectclass=computer))))"
```

Filtering Changes from a Change Log

For connected directories that store changes in a change log, you can use just the following simple operators, which are provided by Oracle Directory Integration and Provisioning, to specify a matching filter for either the Connected Directory Matching Filter (`orclODIPConDirMatchingFilter`) or the OID Matching Filter (`orclODIPOIDMatchingFilter`):

- = (equal operator)
- != (not equal operator)

Note: Connected directories that obtain incremental changes through an LDAP search can also use the preceding operators without the `searchfilter` attribute. However, you can only specify a single expression or the search will fail.

You can use the preceding operators with either LDAP or non-LDAP directories, provided they obtain incremental changes from a change log. Wildcards and pattern matching are not supported with the preceding operators if you do not use the `searchfilter` attribute. However, when multiple operator pairs are including in the filter, the expression is evaluated as a logical AND operation. For example, the following expression includes four operator pairs:

```
"(objectclass=group)(objectclass=organizationalUnit)
(objectclass=user)(objectclass!=computer)"
```

The preceding expression evaluates as follows:

```
objectclass is equal to group
AND objectclass is equal to organizationalUnit
AND objectclass is equal to user
AND objectclass is NOT equal to computer
```

For connected directories that store changes in a change log, a matching filter can synchronize changes for only the attributes that appear in the change log. If you include attributes in a matching filter that do not appear in the change log, the search operation will fail. For this reason, matching filters are of limited use for connected directories that store incremental changes in a change log.

Location and Naming of Files

Table 6–4 tells you where to find the various files used in the directory integration profile and during synchronization.

Table 6–4 Location and Names of Files

File	File Name
Import data file	$*ORACLE_HOME*/ldap/odi/data/import/*Profile_Name*.dat
Export data file	$*ORACLE_HOME*/ldap/odi/data/export/Profile_Name.dat
Additional configuration info file	$*ORACLE_HOME*/ldap/odi/conf/*Profile_Name*.cfg
Mapping rules file	$*ORACLE_HOME*/ldap/odi/conf/*Profile_Name*.map

For example, the name of the data file of the Oracle Human Resources connector is `oraclehrprofile.dat`.

7

Administration of Directory Synchronization

This chapter explains how to manage synchronization profiles. It contains these topics:

- Managing Synchronization Profiles by Using the Oracle Directory Integration and Provisioning Server Administration Tool
- Managing Synchronization Profiles by Using Command-Line Tools

See Also: "Troubleshooting Synchronization" on page C-19

Managing Synchronization Profiles by Using the Oracle Directory Integration and Provisioning Server Administration Tool

This section tells you how to register and deregister a profile by using the Oracle Directory Integration and Provisioning Server Administration tool. It contains these topics:

- Creating a Profile by Using the Oracle Directory Integration and Provisioning Server Administration Tool
- Deleting a Profile by Using the Oracle Directory Integration and Provisioning Server Administration Tool
- Changing the Synchronization Status Attribute

Creating a Profile by Using the Oracle Directory Integration and Provisioning Server Administration Tool

The Oracle Directory Integration and Provisioning Server Administration tool enables you to create a profile in one of two ways:

- By creating a new configuration set entry, then adding a profile to it
- By selecting an existing configuration set entry, then adding a profile to it

To register a directory integration profile:

1. In the navigator pane, select Integration Profile Configuration. The Active Processes box appears in the right pane.
2. From the toolbar, choose **Create**. The Configuration Sets dialog box appears.
3. In the Configuration Sets dialog box, choose **Create**. The Integration Profiles dialog box appears. You have two options:
   - Create an integration profile by copying an existing one

To do this, select the Directory Integration and Provisioning profile you want to copy, then choose **Create Like**. The Integration Profile dialog box displays the **General** tab page.

- Create an integration profile without copying an existing one

  To do this, choose **Create New**. The Integration Profile dialog box displays the **General** tab page.

  See Also: "Integration Profiles" on page A-6 for more information about the Integration Profiles dialog box

4. In the General tab page, fill in the fields.

   The fields in the General tab page are described in Table A–3 on page A-6.

5. Select the Execution tab and fill in the fields.

   The fields in the Execution tab page are described in Table A–4 on page A-7.

6. Select the Mapping tab and fill in the fields.

   The fields in the Mapping tab page are described in Table A–5 on page A-8.

7. Select the Status tab and fill in the fields. Because this page shows the execution status of the connectors, most of the fields are not editable.

   The fields in the Status tab page are described in Table A–6 on page A-8.

8. When you have entered the information, choose **OK**. This returns you to the Configuration Sets dialog box, which now lists the integration profile you just created.

9. Choose **OK** to exit the Configuration Sets dialog box. The profile you created is now registered with Oracle Internet Directory.

Deleting a Profile by Using the Oracle Directory Integration and Provisioning Server Administration Tool

To delete a profile:

1. In the navigator pane, expand **Oracle Internet Directory Servers**, then ***directory server instance***, then **Server Management**, **Directory Integration Server**.

2. Select the configuration set from which to delete the profile. The **Integration Profiles** tab page appears in the right pane.

3. In the **Integration Profiles** tab page, select the profile you want to deregister.

4. Choose **Delete**.

Changing the Synchronization Status Attribute

During synchronization in an export operation, the server constantly updates the synchronization status attribute `orcllastappliedchangenumber`. In the Oracle Directory Integration and Provisioning Server Administration tool, this field is called **OID last applied change number**.

To change this attribute by using the Oracle Directory Integration and Provisioning Server Administration tool:

1. Verify that the Oracle directory integration and provisioning server recognizes the disable flag for the profile.

In the default mode, it can take up to two minutes for the directory integration and provisioning server to recognize this flag. To enable it to recognize this flag sooner, set the refresh interval to a lower value as described in the `odisrv` section in the Oracle Internet Directory server administration tools chapter of the *Oracle Identity Management User Reference*.

2. Disable the agent by using the Oracle Directory Integration and Provisioning Server Administration tool.
3. Make the attribute changes.
4. Re-enable the agent after the change.

Managing Synchronization Profiles by Using Command-Line Tools

You can create, modify, and delete a synchronization profile by using the Directory Integration and Provisioning Assistant as described in the `dipassistant` section of the Oracle Directory Integration and Provisioning tools chapter in the *Oracle Identity Management User Reference*.

8

Bootstrapping of a Directory in Oracle Directory Integration and Provisioning

This chapter discusses directory bootstrapping, which refers to the initial migration of data between a connected directory and Oracle Internet Directory. Because the synchronization process can handle the migration of data between a connected directory and Oracle Internet Directory, you are not required to perform directory bootstrapping. However, relying on the synchronization process to perform the initial migration can be a time consuming process, especially for large amounts of data. For this reason, you should perform directory bootstrapping when you first deploy Oracle Directory Integration and Provisioning.

This chapter contains these topics:

> **See Also:** The chapter on migration of data from other directories and data repositories in the *Oracle Internet Directory Administrator's Guide*

About Directory Bootstrapping in Oracle Directory Integration and Provisioning

In Directory Integration and Provisioning, bootstrapping is handled by using the Directory Integration and Provisioning Assistant with the `bootstrap` option. The command is:

```
dipassistant bootstrap
```

For information about usage of the Directory Integration and Provisioning Assistant, enter:

```
dipassistant bootstrap -help
```

The Directory Integration and Provisioning Assistant enables you to bootstrap by using either a parameter file or a completely configured integration profile. This chapter discusses both approaches.

> **See Also:** The `dipassistant` section of the Oracle Directory Integration and Provisioning tools chapter in the *Oracle Identity Management User Reference*

Bootstrapping by Using a Parameter File

The parameters in this file specify:

- The source and destination data types
- Credentials
- The way the entries need to be mapped between Oracle Internet Directory and the connected directory

The various parameters and the default values that the Directory Integration and Provisioning Assistant assumes for them while reading the file are given in the `dipassistant` section of the Oracle Directory Integration and Provisioning tools chapter in the *Oracle Identity Management User Reference*.

You can bootstrap by using an LDIF file in one of these ways:

- By using the Directory Integration and Provisioning Assistant to read from the source directory
- By using directory-dependent tools to read from the source directory
- By using the Directory Integration and Provisioning Assistant to load data to Oracle Internet Directory

During installation, sample parameter files are copied to the $*ORACLE_HOME*/ldap/odi/samples/ directory. Each file describes the significance of each of the parameters in bootstrapping.

When you run the tools for bootstrapping, be sure that the *`ORACLE_HOME`* and `NLS_LANG` settings are correct.

Bootstrapping can be performed between services with or without one or more intermediate files. However, for large directories, an intermediate LDIF file is required.

This section contains these topics:

- Bootstrapping Without Using an LDIF File
- Bootstrapping by Using an LDIF File

Bootstrapping Without Using an LDIF File

Oracle recommends this method for smaller directories where the entries are:

- Relatively few in number
- In a flat structure
- Not interdependent—that is, the creation of one entry does not depend on the existence of another as, for example, when the creation of a group entry depends on the existence of user member entries

To use this method:

1. Prepare the mapping file with appropriate mapping rules. The mapping file is one of the properties in the bootstrap file. Be sure that it is compatible with the mapping rules defined for synchronization.
2. Create the parameter file with the required details specifying the source as `LDAP` and the destination type as `LDIF`. A sample parameter file, ldp2ldf.properties, is available in $*ORACLE_HOME*/ldap/odi/samples. Make sure that binary attributes are specified as binary in the `SrcAttrType` field.

3. Use the Directory Integration and Provisioning Assistant `bootstrap` command using a configuration file in which:
   - The source is specified as an LDAP directory
   - The destination type is specified as LDIF. Dump the data to an LDIF file.

   Execute the Directory Integration and Provisioning Assistant as follows:

   ```
   Dipassistant bootstrap -cfg parameter_file
   ```

4. Check the `bootstrap.log` and `bootstrap.trc` files for any errors.
5. Use `bulkload` to upload the data to Oracle Internet Directory.
6. For continued synchronization, update the last change number:

   ```
   dipassistant mp -profile profile_name -updcln
   ```

Bootstrapping by Using an LDIF File

This section describes two ways to bootstrap a directory by using an LDIF file.

Bootstrapping from an LDIF File by Using Directory-Dependent Tools to Read the Source Directory

Oracle Corporation recommends that you use this method for large directories. To use this method:

1. Download the data from the directory to an LDIF file. The tool you use depends on the directory from which you are loading the data. If you are bootstrapping from a Microsoft Active Directory, then use "ldifde" to load the data. Be sure to load all the required attributes for each entry.
2. Prepare the mapping file with appropriate mapping rules. When you want to do further synchronization, be sure that the mapping file is the same as the one used for synchronization.
3. Create the parameter file with source and destination as LDIF and other details. A sample parameter file is available in $*ORACLE_HOME*/ldap/odi/samples/ldf2ldf.properties.
4. Use the Directory Integration and Provisioning Assistant `bootstrap` command with a parameter file in which the source is specified as `LDIF` and the destination type is specified as `LDIF`. This converts the source data and creates a new LDIF as required by Oracle Internet Directory. Execute the Directory Integration and Provisioning Assistant as follows:

   ```
   dipassistant bootstrap -cfg parameter_file
   ```

5. Check the `bootstrap.log` and `bootstrap.trc` files for any errors.
6. Use The Oracle Internet Directory bulkload tool (`bulkload.sh`) to upload the data to Oracle Internet Directory.
7. If a corresponding synchronization profile is created for further synchronization, then update the last change number:

   ```
   dipassistant mp -profile profile_name -updcln
   ```

Bootstrapping from an LDIF File by Using the Directory Integration and Provisioning Assistant to Load Data to Oracle Internet Directory

To use this method:

1. Download the data from the directory to an LDIF file. The tool you use depends on the directory from which you are loading the data. If you are bootstrapping from a Microsoft Active Directory, then use "ldifde" to load the data. Be sure to load all the required attributes for each entry.
2. Prepare the mapping file with appropriate mapping rules. When you want to do further synchronization, be sure that the mapping file is the same as the one used for synchronization.
3. Create the properties file with the source specified as `LDIF` and the destination specified as `LDAP`.
4. Use the Directory Integration and Provisioning Assistant `bootstrap` command with a parameter file in which the source is specified as the LDIF file, the destination type is specified as LDAP, and the destination specified as Oracle Internet Directory. This converts the source data and creates entries in Oracle Internet Directory as required. A sample properties file, `ldf2ldp.properties`, is available in $*ORACLE_HOME*/ldap/odi/samples.
5. Check the `bootstrap.log` and `bootstrap.trc` files for any errors.
6. If a corresponding synchronization profile is created for further synchronization, then update the last change number:

   ```
   dipassistant mp -profile profile_name -updcln
   ```

Bootstrapping Directly by Using the Default Integration Profile

Bootstrapping relies on an existing integration profile configured for synchronization. The configuration details are used to connect to the third-party directory.

While using this method, put the source directory in read-only mode.

If the profile is an IMPORT profile, then footprints of the required objects in the connected directory are created in Oracle Internet Directory. If the profile is an EXPORT profile, then footprints of the required objects from Oracle Internet Directory are created in the connected directory.

While creating these entries, the distinguished name and object-level mappings as specified in the integration profile are used. If there is a failure in uploading the entries, then the information is logged in $*ORACLE_HOME*/ldap/odi/log/bootstrap.log. The trace information is written to the file $*ORACLE_HOME*/ldap/odi/log/bootstrap.trc.

For example, for bootstrapping from SunONE Directory Server to Oracle Internet Directory, you would do the following:

1. Customize the default integration profile `IplanetImport`, which is created as part of installation by following the instructions in "Task 1: Configure the Synchronization Profiles for the SunONE Connector" on page 20-4.
2. Enter the following command:

   ```
   dipassistant bootstrap -profile IplanetImport -D  'cn=orcladmin' -w 'welcome'
   ```

3. Check the `bootstrap.log` and `bootstrap.trc` files to be sure that the bootstrapping is successfully completed.

If you are bootstrapping by using the Directory Integration and Provisioning Assistant, then, at the end of the bootstrapping process, the assistant initializes the `lastchangenumber` attribute for further synchronization.

9

Synchronization with Relational Database Tables

This chapter explains how to synchronize data to Oracle Internet Directory from tables in a relational database. The synchronization can be either incremental—for example, one database table row at a time—or all the database tables at once. The process of synchronization with a database server involves executing a directory integration profile. This process has two steps:

1. Retrieving the data from the database. This involves executing a SQL `SELECT` statement that retrieves the specified data records from the database.
2. Writing the data into the directory. This involves converting the retrieved data records to LDAP attribute values and performing the LDAP operation on the directory.

> **Note:** Before reading this chapter, be sure to familiarize yourself with the introductory chapters about Directory Integration and Provisioning—specifically:
>
> - Chapter 1, "Introduction to Oracle Identity Management Integration"
> - Chapter 5, "Oracle Directory Synchronization Service"
>
> Also, be aware that Oracle Internet Directory 10*g* Release 2 (10.1.2) does not enable exporting data from Oracle Internet Directory to a relational database.

This chapter contains these topics:

Preparing the Additional Configuration Information File

During synchronization from a relational database to Oracle Internet Directory, the additional configuration information file governs the retrieval of data from the database. It provides the Oracle directory integration and provisioning server with the following information:

- The SELECT statement to execute
- Either the attribute(s) or the database column(s) to be used in incremental synchronization. Generally, this is either an attribute that contains a timestamp or a change sequence number that the next SQL statement should use to retrieve incremental data.

To configure this file, use the sample file DBReader.cfg.master in the $*ORACLE_HOME*/ldap/odi/conf directory, and edit it to your specifications.

Formatting the Additional Configuration Information File

It is very important to follow the correct format of this file. The various sections are divided using TAG names. Every TAG section has a list of parameters and their respective values. The general layout is as follows.

```
[TAG]
PARAMETER1: value
PARAMETER2: value

[TAG]
PARAMETER1: value
PARAMETER2: value\
VALUE continuation\
value continuation\
end of value continuation

[TAG]
PARAMETER1: value
PARAMETER2: value\
end of value continuation
```

For example, following this format, the DBReader.cfg.master file looks like this:

```
[DBQUERY]
SELECT: SELECT\
EMPNO EmpNum,\
ENAME,\
REPLACE(EMAIL),'@ACME.COM','') UID,\
EMAIL,\
TELEPHONE,\
TO_CHAR(LAST_UPDATE_DATE,'YYYYMMDDHH24MISS') Modified_Date\
FROM\
EMPLOYEE\
WHERE\
LAST_UPDATE_DATE>TO_DATE (:Modified_Date,'YYYYMMDDHH24MISS')\
ORDER BY\
LAST_UPDATE_DATE

[SYNC-PARAMS]
CHANGEKEYATTRS: Modified_Date
```

Note that the entire SELECT statement is put as a value in the parameter SELECT in the section represented by the TAG DBQUERY. Because it is a lengthy value, the value continuation character is put as the last character in every line until the SELECT statement ends.

The CHANGEKEYATTRS parameter value is the name of the column(s) to be used while performing incremental synchronization. The value(s) of these column(s) is always stored in the orclOdipLastAppliedChgNum attribute of the profile. Every time the SELECT statement is executed, the current value(s) of this attribute are put into the

SQL statement accordingly. This ensures that the data is always retrieved incrementally.

If there are multiple column names in the CHANGEKEYATTRS—for example, column1:column2—then the value in the orclOdipLastAppliedChgNum attribute of the profile is stored as value1~value2 and so on, with value1 corresponding to column1 and value2 to column2.

Column names are retrieved into Directory Integration and Provisioning as attribute value pairs and subsequently mapped into LDAP attribute values according to set mapping rules. For this reason, all columns names retrieved in the SELECT statement must be simple names rather than expressions. For example, you can have the expression REPLACE(EMAIL),'@ACME.COM','') but it retrieves the expression value as UID.

In this example, the Modified_Date is the key for incremental synchronization. Because it is a date, it must be represented in a string format.

When the profile is created, the orclOdipLastAppliedChgNum attribute must be set to some value. All changes after this date—that is, rows in the table with LAST_UPDATE_DATE greater than this value— are retrieved. For example, if the orclOdipLastAppliedChgNum attribute is set to 20000101000000, then all employee changes since January 1, 2000 are retrieved.

Because of the ORDER BY clause, all the database rows returned are in the order of LAST_UPDATE_DATE—that is, the changes retrieved and applied to the directory are in chronological order. Once the last change is retrieved and applied:

1. The orclOdipLastAppliedChgNum attribute value is set to the Modified_Date from the last row retrieved.
2. The profile is updated.

Whenever the Directory Integration and Provisioning executes the profile again, it uses the previously stored value.

Preparing the Mapping File

To configure the mapping rules, follow the instructions in "Mapping Rules and Formats" on page 5-2.

Preparing the Directory Integration Profile

You can create the directory integration profile by using the Oracle Directory Integration and Provisioning Server Administration tool or the Directory Integration and Provisioning Assistant. If you use the Oracle Directory Integration and Provisioning Server Administration tool, then you must upload the additional configuration information file and the mapping file by using the Directory Integration and Provisioning Assistant.

To configure the directory integration profile, follow the general instructions in "Registration of Connectors into Oracle Directory Integration and Provisioning" on page 6-1, but with these specific instructions in mind:

- Do not set a value for the Agent Execution Command (orclodipAgentExeCommand) attribute.
- Set the Interface Type (orclodipDataInterfaceType) attribute to DB.

Example: Synchronizing a Relational Database Table to Oracle Internet Directory

This section demonstrates how to synchronize a relational database table to Oracle Internet Directory. It contains these topics:

In this example, the following relational database table containing employee data is synchronized with Oracle Internet Directory.

Table 9–1 Employee Table

EMPNO	ENAME	LAST_UPDATE_DATE	EMAIL	TELEPHONE
98357	JOHN DOE	2-JAN-2000	JOHN.DOE@ACME.COM	435-324-3455
98360	ROGER BECK	3-JUL-2001	ROGER.BECK@ACME.COM	435-324-3600
98365	JIMMY WONG	4-MAR-2001	JIMMY.WONG@ACME.COM	435-324-2390
98370	GEORGE MICHAEL	6-FEB-2002	GEORGE.MICHAEL@ACME.COM	435-324-9232

You can find a sample profile for this example in the directory $*ORACLE_HOME*/ldap/odi/samples. Also present there are the sample configuration and mapping files. In this example:

- The name of the table is `Employee`
- The Profile Name is `TESTDBIMPORT`.
- The employee number (`EMPNO`) is used to JOIN a database record with a directory entry. It is specified in the OID Matching Filter (`orclOdipOIDMatchingFilter`) attribute described in the attributes reference chapter of the *Oracle Identity Management User Reference*.
- This table is present in the `testsync/testsyncpwd` schema in a database. The database is located on the host `machine.acme.com`, the database listener port is `1526` and the SID is `iasdb`. The database URL is `machine.acme.com:1526:iasdb`.
- Appropriate read/write permissions have been given explicitly to this profile, namely,
```
orclodipagentname=testdbimport,
cn=subscriber profile,
cn=changelog subscriber,
cn=oracle internet directory
```
- The profile is created in configuration set 1.

Configuring the Additional Configuration Information File

This example uses the same Additional Configuration Information file described earlier in "Preparing the Additional Configuration Information File" on page 9-1.

Configuring the Mapping File

The mapping file for this example contains the following:

```
DomainRules
NONLDAP:dc=testdbsync,dc=com:uid=%,dc=testdbsync,dc=com
AttributeRules
ename: : : :cn: :person
ename : : : :sn: :person
uid : : : :uid: :inetOrgperson:
EMail: : : :mail: :inetOrgperson
Telephone: : : :telephonenumber: :inetOrgperson
empnum: : : :employeenumber: :inetOrgperson
```

This mapping file specifies the following:

- Directory entries are created as `uid=%,dc=testdbsync,dc=com`. The `%` is a placeholder for the actual value of `uid`. The `uid` must be present in the mapping rules so that it has a value after the mapping. Otherwise the DN construction fails.
- Both the `cn` and `sn` attributes are to have the same value as `ename`.
- The `uid` element must have the value of the `EMail` prefix, which is the element of the e-mail address prior to the '@' character.
- `empnum` becomes `employeenumber` in the directory entry.
- `telephone` becomes `telephone number` in the directory entry.

Configuring the Directory Integration Profile

The directory integration profile for this example contains the attribute values as described in Table 9–2 on page 9-5. A sample integration profile with these values populated and the corresponding mapping and configuration files are available in $*ORACLE_HOME*/ldap/odi/samples directory. You can create the profile by running the Directory Integration and Provisioning Assistant in the `createprofile` mode and specifying the file as the argument. Alternatively, you can create the profile by using the Oracle Directory Integration and Provisioning Server Administration tool.

See Also:

- The `dipassistant` section in the Oracle Directory Integration and Provisioning tools chapter of the *Oracle Identity Management User Reference* for information on the Directory Integration and Provisioning Assistant
- "Creating a Profile by Using the Oracle Directory Integration and Provisioning Server Administration Tool" on page 7-1 for instructions on creating a profile by using the Oracle Directory Integration and Provisioning Server Administration tool

Table 9–2 Directory Integration Profile for TESTDBIMPORT

Attribute	Value
`Profile Name (orclOdipAgentName)`	`TESTDBIMPORT`
Synchronization Mode (`orclOdipSynchronizationMode`)	`IMPORT`
`Professoriats (orclOdipAgentControl)`	`ENABLE`

Table 9–2 (Cont.) Directory Integration Profile for TESTDBIMPORT

Attribute	Value
Agent Execution Command (orclodipAgentExeCommand)	null
Additional Config Info (orclOdipAgentConfigInfo)	As shown in the preceding file. Needs to be uploaded
Connected Directory Account (orclOdipConDirAccessAccount)	testdbsync
Connected Directory Account Password (orclOdipConDirAccessPassword)	testdbsyncpwd
Connected Directory URL (orclOdipConDirURL)	machine.acme.com:1526:iasdb
Interface Type (orclodipDataInterfaceType)	DB
Mapping File:	To be uploaded from a file
OID Matching Filter (orclOdipOIDMatchingFilter)	employeenumber This means that employeenumber is used to search the directory while looking for a match. If a match is found, then the directory entry is modified. Otherwise, a new entry is created. This is necessary to ensure that the orclOdipOIDMatchingFilter attribute is unique in the database also. Once a database row is retrieved, the Oracle directory integration and provisioning server searches the directory for that employeenumber in the domain dc=testdbsync,dc=com according to the domain rules. If it gets a match, it updates that entry with the latest values of the columns in the row retrieved. If it does not get a match, it creates a new entry in the directory with all the attributes from the column values.
Last Applied Change Number (orclodipConDirLastAppliedChgNum)	20000101000000 This means that the first time the profile executes, it retrieves and synchronizes all four rows. Subsequently, it retrieves rows only when the LAST_UPDATE_DATE column in the table is updated to the time last modified.

Uploading the Additional Configuration Information File

Use the Directory Integration and Provisioning Assistant to upload the additional configuration information file, as follows:

```
$ORACLE_HOME/bin/dipassistant modifyprofile [-h hostName] [-p port]
[-D bindDn] [-w password] -profile profName
odip.profile.mapfile=absolute path name of configuration file
```

Uploading the Mapping File

Use the Directory Integration and Provisioning Assistant to upload the mapping file, as follows:

```
$ORACLE_HOME/bin/dipassistant modifyprofile [-h hostName] [-p port]
[-D bindDn] [-w password] -profile profName
```

```
odip.profile.mapfile=absolute path name of mapping file
```

The Synchronization Process

In this example, the sequence of steps in the synchronization process is:

1. The Oracle directory integration and provisioning server starts a new profile thread for the TESTDBIMPORT profile every time the value specified in the scheduling interval (`orclOdipSchedulingInterval`) attribute expires.
2. The profile thread reads the additional configuration information to get the SQL to execute, and then runs the SQL.
3. For every row retrieved from the database, the mapping rules are applied to the record and LDAP attributes are created.
4. Depending on the OID Matching Filter (`orclOdipOIDMatchingFilter`) attribute, the directory integration and provisioning server determines whether a matching entry exists in Oracle Internet Directory or not. If it exists, then it is updated. If not, then a new entry is created. After the directory operation, the last applied change number (`orclodipConDirLastAppliedChgNum`) attribute is updated.

Observations on the Example

When a row is retrieved from the database, it is in the following form:

```
EmpNum: 98357
EName: JOHN DOE
UID: JOHN.DOE
EMAIL: JOHN.DOE@ACME.COM
TELEPHONE: 435-324-3455
Modified_Date: 20000102000000
```

After the mapping is performed on this record, the output is in the following form:

```
dn: uid=john.doe,dc=testdbsync,dc=com
uid: JOHN.DOE
cn: JOHN DOE
sn: JOHN DOE
mail: JOHN.DOE@ACME.COM
employeenumber: 98357
telephonenumber: 435-324-3455
objectclass: person
objectclass: inetorgperson
```

A subtree search is made in the directory with the filter `employeenumber=98357` under the domain `dc=testdbsync,dc=com`. If the search yields an existing entry, then that entry is updated. Otherwise, a new entry is created. Because the OID Matching Filter (`orclOdipOIDMatchingFilter`) attribute is set to `employeenumber`, every database record retrieved must have that column. In this case, it is `EmpNum` as it maps to `employeenumber`.

Any other attributes in the mapping file that are not in the data retrieved by the SQL are ignored—for example, the attribute `birthday`.

After the profile thread processes all the change records from the SQL, it updates the directory with correct values for these attributes:

- Last Applied Change Number (`orclodipConDirLastAppliedChgNum`)
- Last Execution Time (`orclOdipLastExecutionTime`)

- Last Successful Execution Time (orclOdipLastSuccessfulExecutionTime)

10

Synchronization with Oracle Human Resources

If you use Oracle Human Resources as the source of truth for employee data in your enterprise, then you must synchronize between it and Oracle Internet Directory. The Oracle Human Resources connector enables you to do this.

This chapter introduces the Oracle Human Resources connector and explains how to deploy it. It contains these topics:

> **See Also:** Oracle Internet Directory Release Notes to find out which release of Oracle Human Resources can be synchronized with this release of Oracle Internet Directory

Introduction to Synchronization with Oracle Human Resources

The Oracle Human Resources connector enables you to import a subset of employee data from Oracle Human Resources into Oracle Internet Directory. It includes both a prepackaged integration profile and an Oracle Human Resources agent that handles communication with Oracle Internet Directory. You can customize the prepackaged integration profile to meet your deployment needs with either the Oracle Directory Integration and Provisioning Server Administration tool or the Directory Integration and Provisioning Assistant.

You can schedule the Oracle Human Resources connector to run at any time, configuring it to extract incremental changes from the Oracle Human Resources system. You can also set and modify mapping between column names in Oracle Human Resources and attributes in Oracle Internet Directory.

Data that You Can Import from Oracle Human Resources

Table 10–1 lists the tables in the Oracle Human Resources schema. If you choose, you can import most of these attributes into Oracle Internet Directory.

Table 10–1 Tables in Oracle Human Resources Schema

Table Name	Alias Used in the Connector Config Info Field
PER_PEOPLE_F	PER
PER_ADDRESSES	PA
PER_PERIODS_OF_ SERVICE	PPS
PER_PERSON_TYPES	PPT

All of these tables are visible if the login to the Oracle Human Resources database is done with the apps account.

Because attributes can be added or deleted at runtime from the configuration file, the Oracle Human Resources connector dynamically creates a SQL statement that selects and retrieves only the required attributes.

Table 10–2 shows some of the fields in the Oracle Human Resources user interface. These fields appear when you add or modify employee data.

Table 10–2 Fields in the Oracle Human Resources User Interface

ATTRIBUTE NAME	DESCRIPTION	FORM/CANVAS/FIELD_NAME
LAST_NAME	Last name of the person	People/Name/Last
FIRST_NAME	First name of the person	People/Name/First
TITLE	Title of the person	People/Name/Title
SUFFIX	Suffix—for example, Jr, Sr, Ph.D.	People/Name/Suffix
MIDDLE_NAME	Middle name	People/Name/Suffix
SEX	Sex	Gender List box
START_DATE	Hiring date	People/Hire Date
DATE_OF_BIRTH	Date of birth	People/Personal Information/Birth Date
MARITAL_STATUS	Marital status	People/Personal Information/Status
NATIONAL_ INDENTIFIER	Social security number for US residents	People/Identification/Social Security
EMPLOYEE_NUMBER	Employee number	People/Identification/Employee
REGISTERD_ DISABLED_ FLAG	Indicator that the employee has a disability	People/Personal Information/Has Disability
EMAIL_ADDRESS	Electronic mail address	People/Personal Information/EMail
OFFICE_NUMBER	Office location	People/Office Location Info/Office
MAILSTOP	Mail delivery stop	People/Office Location Info/Mail Stop
INTERNAL_ LOCATION	Location	People/Office Location Info/Location
ADDRESS_LINE1	Address line 1	Personal Address Information/Address line 1
ADDRESS_LINE2	Address line 2	Personal Address Information/Address line 2
ADDRESS_LINE3	Address line 3	Personal Address Information/Address line 3
TOWN_OR_CITY	Town or city	Personal Address Information/City

Table 10–2 (Cont.) Fields in the Oracle Human Resources User Interface

ATTRIBUTE NAME	DESCRIPTION	FORM/CANVAS/FIELD_NAME
REGION_1	First region	Personal Address Information/County
REGION_2	Second region	Personal Address Information/State
POSTAL_CODE	Postal code	Personal Address Information/Zip Code
COUNTRY	Country	Personal Address Information/Country
TELEPHONE_ NUMBER_1	First telephone number	Personal Address Information/Telephone
TELEPHONE_ NUMBER_2	Second telephone number	Personal Address Information/Telephone2

Managing Synchronization Between Oracle Human Resources and Oracle Internet Directory

This section contains these topics:

- Task 1: Configure a Directory Integration Profile for the Oracle Human Resources Connector
- Task 2: Configure the List of Attributes to Be Synchronized with Oracle Internet Directory
- Task 3: Configure Mapping Rules for the Oracle Human Resources Connector
- Task 4: Prepare for Synchronization from Oracle Human Resources to Oracle Internet Directory

Task 1: Configure a Directory Integration Profile for the Oracle Human Resources Connector

To configure the prepackaged integration profile that is installed with the Oracle Human Resources connector, you can use either the Oracle Directory Integration and Provisioning Server Administration tool or the Directory Integration and Provisioning Assistant. For information on the Oracle Directory Integration and Provisioning Server Administration tool, see Chapter 7, "Administration of Directory Synchronization". For information on the Directory Integration and Provisioning Assistant, see the `dipassistant` section in the Oracle Directory Integration and Provisioning tools chapter of the Oracle Directory Integration and Provisioning tools chapter in the *Oracle Identity Management User Reference*.

For some of the parameters in the prepackaged integration profile, you must specify values specific to integration with the Human Resources Connector. The parameters specific to the Human Resources Connector are listed in Table 10–3 on page 10-4.

Table 10–3 Attributes Specific to Oracle Human Resources Connector Integration Profile

Attribute	Description
Profile Name (`orclODIPAgentName`)	Unique name by which the connector is identified in the system, used as an RDN component of the DN that identifies the integration profile. The name can contain only alpha-numeric characters. This attribute is mandatory and not modifiable. The default name is `OracleHRAgent`.
Synchronization Mode (`ModeorclODIPSynchronizationMode`)	The direction of synchronization between Oracle Internet Directory and a connected directory. ▪ `IMPORT` indicates importing changes from a connected directory to Oracle Internet Directory. ▪ `EXPORT` indicates exporting changes from Oracle Internet Directory to a connected directory. The default is `IMPORT`. This attribute is mandatory and modifiable. **Note:** In Oracle Internet Directory 10*g* Release 2 (10.1.2), only import operations for Oracle Human Resources are supported.
Execution Information	

Agent Execution Command (`orclODIPAgentExeCommand`)	Connector executable name and argument list used by the directory integration and provisioning server to execute the connector.
	This attribute is mandatory and modifiable.
	The default is:
	`odihragent OracleHRAgent connect=hrdb \`
	`login=%orclodipConDirAccessAccount \`
	`pass=%orclodipConDirAccessPassword \`
	`date=%orclODIPLastSuccessfulExecutionTime \`
	You must set the value in the argument `connect=hrdb` to the connect string of the Oracle Human Resources system database.

Connected Directory Account (`orclodipConDirAccessAccount`)	Valid user account in the connected directory to be used by the connector for synchronization. For the Human Resources Agent, it is a valid user identifier in the Oracle Human Resources database. **See Also:** Chapter 10, "Synchronization with Oracle Human Resources" for typical usage of passing it in the command-line
Additional Config Info (`orclODIPAgentConfigInfo`)	Any configuration information that you want the connector to store in Oracle Internet Directory. It is passed by the directory integration and provisioning server to the connector at time of connector invocation. The information is stored as an attribute and the directory integration and provisioning server does not have any knowledge of its content. The value stored in this attribute represents (for Oracle Human Resources connector) all attributes that need to be synchronized from Oracle Human Resources. **See Also:** "Task 2: Configure the List of Attributes to Be Synchronized with Oracle Internet Directory" on page 10-5 This attribute is mandatory for the Oracle Human Resources connector, and modifiable by editing the configuration file and uploading it again into the profile. You cannot modify this attribute by using the Oracle Directory Integration and Provisioning Server Administration tool.
Connected Directory URL	The host and port details of the connected directory. It must be entered in this format: *`host`*:*`port`*:*`sid`*.

Table 10–3 (Cont.) Attributes Specific to Oracle Human Resources Connector Integration Profile

Attribute	Description
Interface Type (orclODIPInterfaceType)	The interface used for data transfer. Since it is in the form of a tagged file, it is set to TAGGED. **Note:** You should not modify this attribute for Oracle Human Resources Profile.
Mapping Information	
Mapping Rules (orclODIPAttributeMappingRules)	Attribute for storing the mapping rules. Store the mapping rules in a file by using the Directory Integration and Provisioning Assistant. This attribute is mandatory for Oracle Human Resources and is modifiable. **See Also:** ■ "Mapping Rules and Formats" on page 5-2 ■ "Configuring Mapping Rules" on page 6-3
Connected Directory Matching Filter (orclODIPConDirMatchingFilter)	This is not used in Oracle Human Resources connectivity.
OID Matching Filter (orclODIPOIDMatchingFilter)	This attribute names an LDAP filter that is used to search for a target entry in Oracle Internet Directory. The Oracle directory integration and provisioning server uses this filter to find out what kind of LDAP operation it needs to do to synchronize. It is of the form employeenumber=% It is optional and modifiable.
Status Information	
OID Last Applied Change Number (orcllastappliedChangenumber)	This attribute, standard for all EXPORT profiles, does not apply to Oracle Human Resources synchronization.
Last Applied Change Number (orclODIPConDirLastAppliedChgNum)	This attribute, standard for all profiles, does not apply to the Oracle Human Resources synchronization.

Task 2: Configure the List of Attributes to Be Synchronized with Oracle Internet Directory

The default Oracle Human Resources profile provides a default list of attributes to be synchronized from Oracle Human Resources to Oracle Internet Directory. You can customize this list, adding attributes to it or removing attributes from it.

The default attribute list is stored in the orclodipAgentConfigInfo attribute as part of the integration profile.The configuration information is also available in the file oraclehragent.cfg.master that is located under the $*ORACLE_HOME*/ldap/odi/conf directory.

> **Note:** Do not modify the oraclehragent.cfg.master file; it serves as a backup.

The columns in the default list of Oracle Human Resources attributes are:

Table 10–4 Oracle Human Resources Attributes Synchronized with Oracle Internet Directory by Default

Column	Description
ATTRNAME	The output tag generated in the output data file

Table 10–4 (Cont.) Oracle Human Resources Attributes Synchronized with Oracle Internet Directory by Default

Column	Description
COLUMN_NAME	Database column name from where to obtain this value
TABLE_NAME	Database table name from where to obtain this value
FORMAT	The column data type of this attribute. (ASCII, NUMBER, DATE)
MAP	Indicator of whether to extract this attribute from Oracle Human Resources or not. A value of `Y` indicates that it will be extracted and a value of `N` indicates that it will not be.

The `oraclehragent.cfg.master` file contains the following:

```
ATTRNAME:COLUMN_NAME:TABLE_NAME:FORMAT:MAP
PersonId:person_id:PER:NUMBER:Y
PersonType:person_type_id:PER:NUMBER:Y
PersonTypeName:system_person_type:PPT:ASCII:Y
LastName:last_name:PER:ASCII:Y
StartDate:start_date:PER:DATE:Y
BirthDate:date_of_birth:PER:DATE:Y
EMail:email_address:PER:ASCII:Y
EmployeeNumber:employee_number:PER:NUMBER:Y
FirstName:first_name:PER:ASCII:Y
FullName:full_name:PER:ASCII:Y
knownas:known_as:PER:ASCII:Y
MaritalStatus:marital_status:PER:ASCII:Y
middleName:middle_names:PER:ASCII:Y
country:country:PA:ASCII:Y
socialsecurity:national_identifier:PER:ASCII:Y
Sex:sex:PER:ASCII:Y
Title:title:PER:ASCII:Y
suffix:suffix:PER:ASCII:Y
street1:address_line1:PA:ASCII:Y
zip:postal_code:PA:ASCII:Y
Address1:address_line1:PA:ASCII:Y
Address2:address_line2:PA:ASCII:Y
Address3:address_line3:PA:ASCII:Y
TelephoneNumber1:telephone_number_1:PA:ASCII:Y
TelephoneNumber2:telephone_number_2:PA:ASCII:Y
TelephoneNumber3:telephone_number_3:PA:ASCII:Y
town_or_city:town_or_city:PA:ASCII:Y
state:region_2:PA:ASCII:Y
Start_date:effective_start_date:PER:DATE:Y
End_date:effective_end_date:PER:DATE:Y
per_updateTime:last_update_date:PER:DATE:Y
pa_updateTime:last_update_date:PA:DATE:Y
```

Modifying Additional Oracle Human Resources Attributes for Synchronization

To include additional Oracle Human Resources attributes for synchronization, follow these steps:

1. Copy the `oraclehragent.cfg.master` file and name it anything other than *`Agent_Name`*`.cfg`. This is because the directory integration and provisioning server generates a configuration file with that name, using it to pass the configuration information to the Oracle Human Resources agent at run time.

2. Include an additional Oracle Human Resources attribute for synchronization by adding a record to this file. To do this, you need this information:

   - Table name in the database from which the attribute value is to be extracted. These tables are listed in Table 10–1 on page 10-2. The file uses abbreviated names for the four tables used in the synchronization.
   - Column name in the table
   - Column datatype. Valid values are ASCII, NUMBER, DATE

   You also need to assign an attribute name to the column name. This acts as the output tag that is used to identify this attribute in the output file. This tag is used in the mapping rules to establish a rule between the Oracle Human Resources attribute and the Oracle Internet Directory attribute.

   You must also ensure that the `map` column—that is, the last column in the record—is set to the value `Y`.

   Note: If you add a new attribute in the attribute list, then you must define a corresponding rule in the `orclodipAttributeMappingRules` attribute. Otherwise the Oracle Human Resources attribute is not synchronized with the Oracle Internet Directory even if it is being extracted by the Oracle Human Resources connector.

Excluding Oracle Human Resources Attributes from Synchronization

To exclude an Oracle Human Resources attribute that is currently being synchronized with Oracle Internet Directory:

1. Copy the `oraclehragent.cfg.master` file and name it anything other than *`Agent_Name`*`.cfg`. This is because the directory integration and provisioning server generates a configuration file with that name, using it to pass the configuration information to the Oracle Human Resources connector at run time.
2. Do one of the following:

   - Comment out the corresponding record in the attribute list by putting a hash sign (#) in front of it
   - Set the value of the column `map` to `N`

Configuring a SQL SELECT Statement in the Configuration File to Support Complex Selection Criteria

If the previous supporting attribute configuration is not sufficient to extract data from the Oracle Human Resources database, then the Oracle Human Resources agent also supports execution of a pre configured SQL `SELECT` statement in the configuration file. There is a TAG to indicate this in the configuration file, namely, a `[SELECT]` in the configuration file.

The following example shows a sample select statement to retrieve some information from the Oracle Human Resources database. Note that only the SQL statement should follow the `[SELECT]` Tag. The `BINDVAR` Bind Variable needs to be there to retrieve incremental changes. The substitutes passes this value (the time stamp) to the Oracle Human Resources connector.

All the columns expressions retrieved in the `SELECT` statement must have column names—for example, `REPLACE(ppx.email_address),'@ORACLE.COM',''` is

retrieved as EMAILADDRESS. The Oracle Human Resources connector writes out EMAILADDRESS as the attribute name in the output file with its value as the result of the expression REPLACE(ppx.email_address),'@ORACLE.COM''').

The following is an example of a a SELECT statement in a configuration file.

```
[SELECT]

SELECT
     REPLACE(ppx.email_address),'@ORACLE.COM',''), EMAILADDRESS ,
     UPPER(ppx.attribute26) GUID,
     UPPER(ppx.last_name) LASTNAME,
     UPPER(ppx.first_name) FIRSTNAME,
     UPPER(ppx.middle_names) MIDDLENAME,
     UPPER(ppx.known_as) NICKNAME,
     UPPER(SUBSTR(ppx.date_of_birth,1,6)) BIRTHDAY,
     UPPER(ppx.employee_number) EMPLOYEEID,
     UPPER(ppos.date_start)  HIREDATE,
FROM
   hr_organization_units hou,
   per_people_x ppx,
   per_people_x mppx,
   per_periods_of_service ppos
  WHERE
      pax.supervisor_id  = mppx.person_id(+)
  AND pax.organization_id = hou.organization_id(+)
  AND ppx.person_id = ppos.person_id
  AND ppx.person_id = pax.person_id
  AND ppos.actual_termination_date IS NULL
  AND UPPER(ppx.current_employee_flag) = 'Y'
  AND ppx.last_update_date >= (:BINDVAR,'YYYYMMDDHH24MISS')
```

Task 3: Configure Mapping Rules for the Oracle Human Resources Connector

Attribute mapping rules govern how the directory integration and provisioning server converts attributes between Oracle Human Resources and Oracle Internet Directory. You can customize the mapping rules you want the directory integration and provisioning server to use.

The Oracle Human Resources agent profile has a default mapping file with a set of mapping rules in the attribute orclodipAttributeMappingRules. This information is also stored in the file named oraclehragent.map.master located under the $*ORACLE_HOME*/ldap/odi/conf directory.

Note: Do not modify the oraclehragent.map.master file. It serves as a backup.

See Also: "Mapping Rules and Formats" on page 5-2 for the contents of the oraclehragent.map.master and a description of the format of the mapping rules records

Task 4: Prepare for Synchronization from Oracle Human Resources to Oracle Internet Directory

This section explains how to set up synchronization from Oracle Human Resources to Oracle Internet Directory.

Preparing for Synchronization

To prepare for synchronization between Oracle Human Resources and Oracle Internet Directory, follow these steps:

1. Ensure that the Oracle Human Resources connector and the directory integration and provisioning server are installed on the host from which you want to run the Oracle Human Resources connector.

   See Also: The file `install.txt` and the Release Notes for Oracle Internet Directory 10*g* Release 2 (10.1.2) for more details

2. Ensure that you have the information for accessing the Oracle Human Resources system, including:
   - Connect string to the Oracle Human Resources system database
   - Access account
   - Password
3. Configure an integration profile for the Oracle Human Resources connector, as described in "Task 1: Configure a Directory Integration Profile for the Oracle Human Resources Connector" on page 10-3. Ensure that all values in the integration profile are properly set, including:
   - Oracle Human Resources attribute list
   - Oracle Human Resources attribute mapping rules
   - Scheduling interval
4. Once everything is properly set, set the Profile Status (`orclodipagentcontrol`) attribute to `ENABLE`. This indicates that the Oracle Human Resources connector is ready to run.
5. Start the Oracle directory server and the Oracle Human Resources system if they are not already running on the respective hosts.
6. When everything is ready, start the directory integration and provisioning server if it is not already running on this host.

   See Also: "Starting, Stopping, and Restarting the Oracle Directory Integration and Provisioning Server" on page 4-8 for instructions about starting and stopping the directory integration and provisioning server

The Synchronization Process

Once the Oracle Human Resources system, Oracle Internet Directory, and the directory integration and provisioning server are running and the Oracle Human Resources connector is enabled, the directory integration and provisioning server automatically starts synchronizing changes from the Oracle Human Resources system into Oracle Internet Directory. It follows this process:

1. Depending on the value specified in the Last Execution Time (`orclodipLastExecutionTime`) and the Scheduling Interval (`orclodipschedulinginterval`), the directory integration and provisioning server invokes the Oracle Human Resources connector.
2. The Human Resources agent extracts:

- All the changes from the Oracle Human Resources System based on the time specified in the `orclodipLastSuccessfulExecutionTime` attribute in the integration profile
- Only the attributes specified in the `orclodipAgentConfigInfo` attribute in the profile

It then writes the changes into the Oracle Human Resources import file, namely $*ORACLE_HOME*/ldap/odi/import/HR_Agent_Name.dat**.**

3. After the agent completes the execution, it creates a data file that looks something like the following:

```
FirstName: John
LastName: Liu
EmployeeNumber: 12345
Title: Mr.
Sex: M
MaritalStatus: Married
TelephoneNumber: 123-456-7891
Mail: Jliu@my_company.com
Address: 100 Jones Parkway
City: MyTown
```

4. The Oracle directory integration and provisioning server imports the changes to Oracle Internet Directory by doing the following:
   - Reading each change record from the import file
   - Converting each change record into an LDAP change entry based on the rules specified in the Mapping Rules (`orclodipAttributeMappingRules`) in the integration profile.
5. After importing all the changes to Oracle Internet Directory, Oracle Human Resources connector moves the import file to the archive directory, $*ORACLE_HOME*/ldap/odi/import/archive. The status attributes Last Execution Time (`orclodipLastExecutionTime`) and Last Successful Execution Time (`orclodipLastSuccessfulExecutionTime`) are updated to the current time.

   If the import operation fails, only the Last Execution Time (`orclodipLastExecutionTime`) attribute is updated, and the connector once again attempts to extract the changes from Human Resources system based on the Last Successful Execution Time (`orclodipLastSuccessfulExecutionTime`) attribute. The reason for failure is logged in the trace file in $*ORACLE_HOME*/ldap/odi/HR_Agent_Name.trc file.

Bootstrapping Oracle Internet Directory from Oracle Human Resources

There are two ways to bootstrap Oracle Internet Directory from Oracle Human Resources:

- Use the Oracle Human Resources connector. In the integration profile, set the `orclodipLastSuccessfulExecutionTime` to a time before Oracle Human Resources was installed.
- Use external tools to migrate data from Oracle Human Resources into Oracle Internet Directory

11

Synchronization with Third-Party Metadirectory Solutions

To enable synchronization with supported third-party metadirectory solutions, Oracle Internet Directory uses change logs. The Oracle directory integration and provisioning server does not provide mapping or scheduling services for third-party metadirectory solutions.

This chapter describes how change log information is generated and how supporting solutions use that information. It tells you how to enable third-party metadirectory solutions to synchronize with Oracle Internet Directory.

This chapter contains these topics:

About Change Logs

Oracle Internet Directory records each change as an entry in the change log container. A third-party metadirectory solution retrieves changes from the change log container and applies them to the third-party directory. To retrieve these changes, the third-party metadirectory solution must subscribe to the Oracle Internet Directory change logs.

Each entry in the change log store has a change number. The third-party metadirectory solution keeps track of the number of the last change it applied, and it retrieves from Oracle Internet Directory only those changes with numbers greater than the last change it applied. For example, if the last change a third-party metadirectory solution retrieved had a number of 250, then subsequent changes it retrieves would have numbers greater than 250.

Note: If a third-party metadirectory solution is not subscribed to the Oracle Internet Directory change logs, and the first change it retrieves is more than one number higher than the last change it last applied, then some of the changes in the Oracle Internet Directory change log have been purged. In this case, the third-party metadirectory solution must read the entire Oracle Internet Directory to synchronize its copy with that in Oracle Internet Directory.

See Also: "Components Involved in Oracle Directory Synchronization" on page 5-1 for a conceptual discussion of directory integration profiles

Enabling Third-Party Metadirectory Solutions to Synchronize with Oracle Internet Directory

To enable third-party metadirectory solutions to retrieve changes from Oracle Internet Directory, perform the tasks described in this section.

- Task 1: Perform Initial Bootstrapping
- Task 2: Create a Change Subscription Object in Oracle Internet Directory for the Third-Party Metadirectory Solution

Task 1: Perform Initial Bootstrapping

To bootstrap a directory to synchronize data between a local directory and Oracle Internet Directory, do the following:

1. Find the number of the last change recorded in Oracle Internet Directory. This number is contained in the DSE root attribute, `lastChangeNumber`.

   To find the number of the last change recorded in Oracle Internet Directory, use ldapsearch. Enter the following command:

   ```
   ldapsearch -h host_name -p port_number -s base -b "" 'objectclass=*'
   lastchangenumber
   ```

   If the change log does not contain change entries because they have been purged, then the last change number retrieved is `0` (zero).

2. Use ldifwrite to export data from Oracle Internet Directory into an LDIF file.
3. Convert the LDIF file to a format suitable to the client directory, then load it into the client directory.

Note: Initial bootstrapping is not required with a new installation of Oracle Internet Directory. In this case, the current change number of the newly installed Oracle Internet Directory is `0` (zero).

See Also: See the `ldifwrite` section in the Oracle Internet Directory data management tools chapter of the *Oracle Identity Management User Reference*

Task 2: Create a Change Subscription Object in Oracle Internet Directory for the Third-Party Metadirectory Solution

To enable a third-party metadirectory solution to synchronize with Oracle Internet Directory, you must create a change subscription object for it in Oracle Internet Directory. This gives the third-party metadirectory solution access to change log objects stored in Oracle Internet Directory.

About the Change Subscription Object

The change subscription object is an entry located under the following container in Oracle Internet Directory:

```
cn=Subscriber Profile,cn=ChangeLog Subscriber,cn=Oracle Internet Directory
```

This change subscription object provides a unique credential for a third-party metadirectory solution to bind with Oracle Internet Directory and to retrieve changes from it. You associate the change subscription object with the auxiliary object class `orclChangeSubscriber`. This object class has several attributes, of which the following are mandatory:

- `userPassword`

  Password to be used by the directory when accessing the change log object in Oracle Internet Directory

- `orclLastAppliedChangeNumber`

  Number of the change applied during the last synchronization. This attribute allows the directory to retrieve only the changes in Oracle Internet Directory it has not already applied.

Creating a Change Subscription Object

To create a change subscription object, use ldapadd. The following example uses an input file, named `add.ldif`, to create and enable a change subscription object, named `my_change_subscription_object`, under the container `cn=Subscriber Profile,cn=ChangeLog Subscriber,cn=Oracle Internet Directory`. The `orclLastAppliedChangeNumber` is the current change number in the directory before initial bootstrapping—in this example, 250.

- Edit file `add.ldif`:

```
dn: cn=my_change_subscription_object,cn=Subscriber Profile,
  cn=ChangeLog Subscriber,cn=Oracle Internet Directory
userpassword: my_password
orclLastAppliedChangeNumber: 250
orclSubscriberDisable: 0
objectclass: orclChangeSubscriber
objectclass: top
```

- Add the entry:

```
ldapadd -h my_host -p 389 -f add.ldif
```

> **See Also:** "Disabling and Deleting Change Subscription Objects" on page 11-4 for instructions on temporarily disabling change subscription objects or deleting them altogether

The Synchronization Process

This section contains these topics:

- How a Connected Directory Retrieves Changes the First Time from Oracle Internet Directory
- How a Connected Directory Updates the orclLastAppliedChangeNumber Attribute in Oracle Internet Directory

How a Connected Directory Retrieves Changes the First Time from Oracle Internet Directory

In this example, a connected directory with a change subscription object named `my_change_subscription_object` acquires changes from Oracle Internet Directory.

```
ldapsearch -h my_host -p 389 -b "cn=changeLog" -s one
(&(objectclass=changeLogEntry)
(changeNumber >= orclLastAppliedChangeNumber )
( ! (modifiersname =cn=my_change_subscription_object,cn=Subscriber Profile,
     cn=ChangeLog Subscriber,cn=Oracle Internet Directory ) ) )
```

When the directory is retrieving changes for the first time, the value for `orclLastAppliedChangeNumber` is the number you set in "Task 2: Create a Change Subscription Object in Oracle Internet Directory for the Third-Party Metadirectory Solution" on page 11-2.

The argument `(!(modifiersname=`*client_bind_dn*`))` in the filter ensures that Oracle Internet Directory does not return changes made by the connected directory itself.

How a Connected Directory Updates the orclLastAppliedChangeNumber Attribute in Oracle Internet Directory

After retrieving changes from Oracle Internet Directory, the connected directory updates the `orclLastAppliedChangeNumber` attribute in its change subscription object in Oracle Internet Directory. This allows Oracle Internet Directory to purge changes that connected directories have already applied. It also enables the connected directory to retrieve only the most recent changes, ignoring those it has already applied.

This example uses an input file, `mod.ldif`, in which the connected directory has a change subscription object named `my_change_subscription_object`, and the last applied change number is 121. The connected directory updates `orclLastAppliedChangeNumber` in its change subscription object in Oracle Internet Directory as follows:

1. Edit `mod.ldif`:

```
dn: cn=my_change_subscription_object,cn=Subscriber Profile,
    cn=ChangeLog Subscriber,cn=Oracle Internet Directory
changetype:modify
replace: orclLastAppliedChangeNumber
orclLastAppliedChangeNumber: 121
```

2. Use ldapmodify to load the edited `mod.ldif` file:

```
ldapmodify -h host -p port -f mod.ldif
```

> **See Also:** The chapter on garbage collection in *Oracle Internet Directory Administrator's Guide* for information about purging changes according to change numbers

Disabling and Deleting Change Subscription Objects

You can temporarily disable an existing change subscription object, or delete it altogether. This section contains these topics:

- Disabling a Change Subscription Object
- Deleting a Change Subscription Object

Disabling a Change Subscription Object

If a change subscription object already exists for a third-party metadirectory solution, but you want to disable it temporarily, then set the `orcl`SubscriberDisable attribute to

1. The following example uses an input file, `mod.ldif`, to disable a change subscription object.

- Edit file `mod.ldif`:

```
dn: cn=my_change_subscription_object,cn=Subscriber Profile,
    cn=ChangeLog Subscriber,cn=Oracle Internet Directory
changetype: modify
replace: orclSubscriberDisable
orclSubscriberDisable: 1
```

- Modify the entry:

```
ldapmodify -h my_ldap_host -p 389 -v -f mod.ldif
```

Deleting a Change Subscription Object

To delete a change subscription object, use ldapdelete. Enter the following command:

```
ldapdelete -h ldap_host -p ldap_port
           "cn=my_change_subscription_object,cn=Subscriber Profile,
            cn=ChangeLog Subscriber,cn=Oracle Internet Directory"
```

Part IV

Provisioning in Oracle Identity Management

This part discusses the concepts and components involved in provisioning, the process through which an application receives changes to user or group entries or attributes that it needs to track. It contains these chapters:

12

Oracle Provisioning Service Concepts

This chapter discusses the Oracle Provisioning Service, which is used to provision users in Oracle Identity Management. It contains these sections:

See Also:

- The chapter on developing provisioning-integrated applications in *Oracle Identity Management Application Developer's Guide*
- "Troubleshooting Provisioning" on page C-14

What is Provisioning?

Provisioning refers to the process of providing users, groups, and other objects with access to applications and other resources that may be available in an enterprise environment. A provisioning-integrated application refers to an application that has registered for provisioning events and registered a provisioning-integration profile in Oracle Internet Directory. At times, you may want to synchronize all user entries in an application-specific directory with those in Oracle Internet Directory, but provision a particular application to receive notification about only some of them. For example, the directory for Oracle Human Resources typically contains data for all employees in an enterprise, and you would probably want to synchronize all of that data with Oracle Internet Directory. However, you might want to provision another application, such as Oracle Email, to be notified only when members join or leave a particular group.

Before a user account can be provisioned for applications in an Oracle Identity Management deployment, it must first be created in Oracle Internet Directory. User accounts can be created in Oracle Internet Directory with any of the following tools or methods:

- Oracle Internet Directory Provisioning Console
- The Directory Integration and Provisioning Assistant's `bulkprov` operation

- Synchronization with third-party directories
- Command-line LDAP tools

The Oracle Provisioning Service can be invoked for any user entries, regardless of how they were created in Oracle Internet Directory. However, simply creating a user entry in Oracle Internet Directory does not necessarily mean that the user entry will have access to all applications in the Oracle Identity Management environment. The user account must be manually provisioned by an administrator or automatically provisioned according to an application's provisioning policies. An application's default provisioning policy can be one of the following:

- Provision all users
- Do not provision users
- Provision users after evaluating a provisioning policy

Provisioning policies are entirely dependent on the needs and requirements within each enterprise environment. For example, an organization may choose to provision all users with access to an e-mail application, but may restrict the users that are provisioned to access a human resources application.

Components of the Oracle Provisioning Service

The Oracle Provisioning Service consists of the following components:

- The Oracle directory integration and provisioning server.

  See Also: Chapter 4, "Managing the Oracle Directory Integration and Provisioning Server"

- The Oracle Internet Directory Provisioning Console, a ready-to-use standalone application created by using Oracle Delegated Administration Services. The Provisioning Console works closely with the Oracle Directory Integration and Provisioning Administration Tools.

  Note:

  - Chapter 14, "Managing with the Provisioning Console"
  - *Oracle Identity Management Guide to Delegated Administration*

- A provisioning integration profile for each provisioning-integrated application in which you want to provision users. You create a provisioning-integration profile by using the Provisioning Subscription Tool.

  See Also: The `oidprovtool` section in the Oracle Directory Integration and Provisioning tools chapter of the *Oracle Identity Management User Reference* for information about the Provisioning Subscription Tool

Understanding Provisioning Concepts

This section explains how applications are provisioned with the Oracle Provisioning Service. It contains these topics:

Synchronous Provisioning

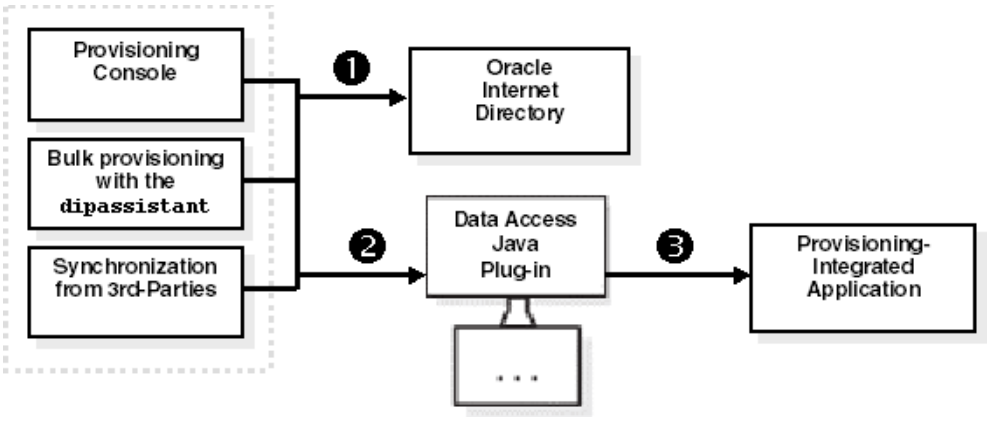
A provisioning-integrated application can maintain user information in Oracle Internet Directory or a third-party repository. Applications that maintain user information in Oracle Internet Directory can use the Data Access Java plug-in to create, modify, and delete user entries whenever the change occurs in Oracle Internet Directory.

> **See Also:** *Oracle Identity Management Application Developer's Guide* for more information on the Data Access Java plug-in

The Data Access Java plug-in can be invoked directly from Oracle Identity Management, including the Provisioning Console, bulk provisioning with the Directory Integration and Provisioning Assistant, and command-line LDAP tools. For this reason, applications that can be provisioned with the Data Access Java plug-in are provisioned synchronously; no separate provisioning event needs to be sent to the application from the Oracle directory integration and provisioning server. The Data Access Java plug-in returns an execution status of `SUCCESS` or `FAILURE` to the Oracle directory integration and provisioning server. If an execution status of `SUCCESS` is returned for the Data Access Java plug-in, then a provisioning status is also returned, which is recorded in user's provisioning status attribute in Oracle Internet Directory for the specific provisioning-integrated application. If an execution status of FAILURE is returned for new user provisioning requests, then the user's provisioning status is assigned a value of `PROVISIONING_FAILURE`. See "Provisioning Status in Oracle Internet Directory" on page 12-10 for a list of provisioning statuses.

Figure 12–1 illustrates the process of how an application is synchronously provisioned from the Provisioning Console, bulk provisioning with the Directory Integration and Provisioning Assistant, and from third-party directories.

Figure 12–1 Synchronous Provisioning Process



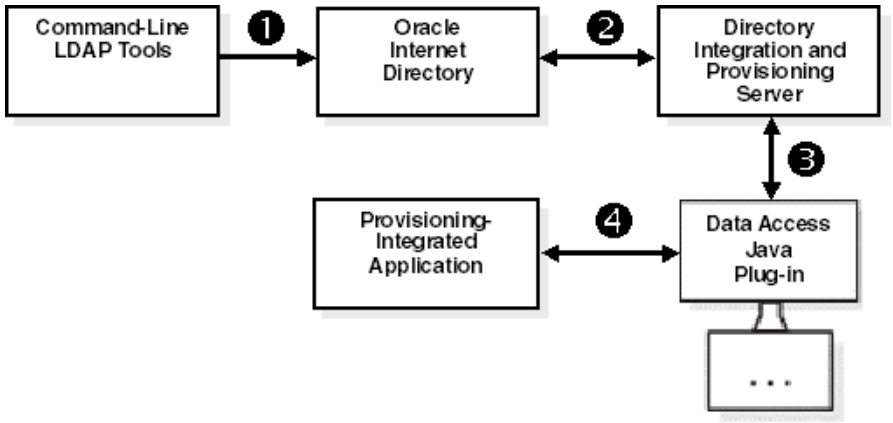
As illustrated in Figure 12–1, synchronous provisioning with the Oracle Provisioning Service from the Provisioning Console, bulk provisioning with the Directory

Integration and Provisioning Assistant, and from third-party directories follows this process:

1. A new user entry is created in Oracle Internet Directory from one of the following sources:
   - Oracle Internet Directory Provisioning Console
   - Bulk provisioning with the Directory Integration and Provisioning Assistant
   - Synchronization with third-party directories
2. The Oracle Identity Management component that created the new user entry invokes the Data Access Java plug-in.
3. The Data Access Java plug-in provisions the new user account in the application.

Figure 12–2 illustrates the process of how an application is synchronously provisioned using command-line LDAP tools.

Figure 12–2 Synchronous Provisioning from Command-Line LDAP Tools



As illustrated in Figure 12–2, synchronous provisioning from command-line LDAP tools follows this process:

1. A command-line LDAP tool creates a new user entry in Oracle Internet Directory.
2. At the next scheduled synchronization interval, the Oracle directory integration and provisioning server identifies new users entries in Oracle Internet Directory that require provisioning.
3. The Oracle directory integration and provisioning server invokes the Data Access Java plug-in.
4. The Data Access Java plug-in provisions the new user accounts in the application.

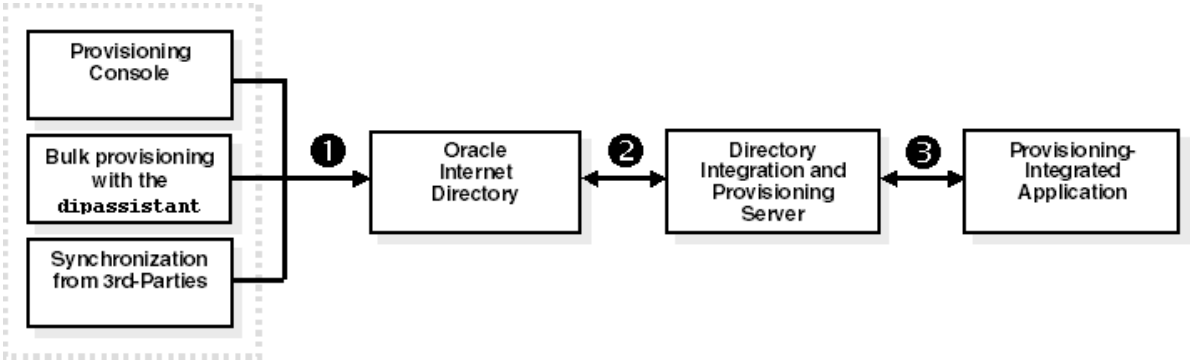
Asynchronous Provisioning

The Oracle directory integration and provisioning server propagates PL/SQL events to a provisioning-integrated application, which then executes a PL/SQL plug-in to process the events. Execution of a PL/SQL plug-in occurs within the application repository and not within the address space of any Oracle Identity Management components. Because, provisioning is handled by a PL/SQL plug-in and not by any components of Oracle Identity Management, provisioning-integrated applications that implement a PL/SQL plug-in are provisioned asynchronously. The PL/SQL plug-in returns an execution status of `SUCCESS` or `FAILURE` to the Oracle directory integration

and provisioning server. If an execution status of SUCCESS is returned for the PL/SQL plug-in, then a provisioning status is also returned, which is recorded in the user's provisioning status attribute in Oracle Internet Directory for the specific provisioning-integrated application. If an execution status of FAILURE is returned for new user provisioning requests, then the user's provisioning status is assigned a value of PROVISIONING_FAILURE. See "Provisioning Status in Oracle Internet Directory" on page 12-10 for a list of provisioning statuses.

Figure 12–3 illustrates the process of how an application is asynchronously provisioned from the Provisioning Console, by using bulk provisioning with the Directory Integration and Provisioning Assistant, or from third-party directories.

Figure 12–3 Asynchronous Provisioning Process

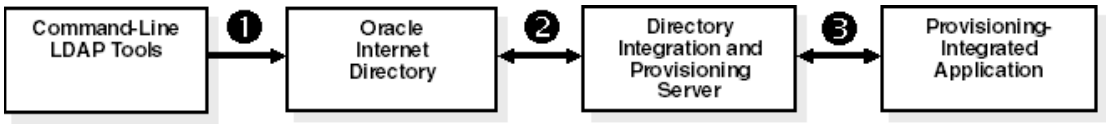


As illustrated in Figure 12–3, asynchronous provisioning from the Provisioning Console, bulk provisioning with the Directory Integration and Provisioning Assistant, and third-party directories follows this process:

1. A new user entry and an associated entry containing application-specific user preferences are created in Oracle Internet Directory from one of the following sources:
   - Oracle Internet Directory Provisioning Console
   - Bulk provisioning with the Directory Integration and Provisioning Assistant
   - Synchronization with third-party directories
2. At the next scheduled synchronization interval, the Oracle directory integration and provisioning server identifies new users entries in Oracle Internet Directory that require provisioning.
3. Provisioning events are sent from the Oracle directory integration and provisioning server to the PL/SQL plug-in.

Figure 12–4 illustrates the process of how an application is asynchronously provisioned using command-line LDAP tools.

Figure 12–4 Asynchronous Provisioning using Command-Line LDAP Tools



As illustrated in Figure 12–4, asynchronous using command-line LDAP tools follows this process:

1. A new user entry is created in Oracle Internet Directory from one of the following sources using a command-line LDAP tool.
2. At the next scheduled synchronization interval, the Oracle directory integration and provisioning server identifies new users entries in Oracle Internet Directory that require provisioning and creates an associated entry containing application-specific user preferences.
3. Provisioning events are sent from the Oracle directory integration and provisioning server to the PL/SQL plug-in.

Provisioning Data Flow

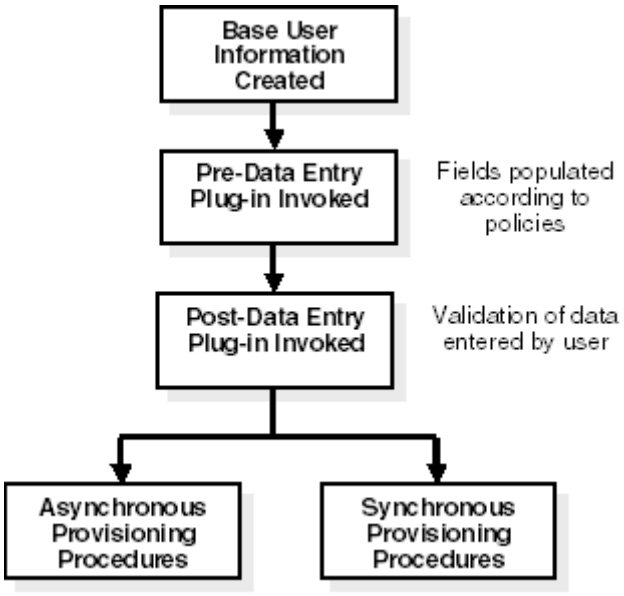
Regardless of whether it is provisioned synchronously or asynchronously, an application can invoke the Pre-Data Entry and Post-Data Entry plug-ins to enhance provisioning intelligence and implement business policies. Both plug-ins are invoked by Oracle Identity Management components such as the Oracle Internet Directory Provisioning Console and bulk provisioning with the Directory Integration and Provisioning Assistant.

The Pre-Data Entry plug-in populates fields according to provisioning policies. The primary purpose of this plug-in is to determine whether a user should be provisioned in a given application. For example, if an organization has a a policy where only managers are provisioned for a financial application, the Pre-Data Entry plug-in can be used to identify which user entries to provision. Common user attributes are already populated when this plug-in is invoked, so it should have adequate information to make provisioning decisions.

The Post-Data Entry plug-in primarily validates data entered by users for common attributes and application-specific attributes. The validation for the plug-in must be successful in order for provisioning to continue.

Figure 12–5 illustrates the provisioning data flow using the Pre-Data Entry and Post-Data Entry plug-ins.

Figure 12–5 Provisioning Data Flow



As illustrated in Figure 12–5, the provisioning data flow follows this process:

1. Base user information is created.
2. The Pre-Data Entry plug-in is invoked, which populates fields according to policies.
3. The Post-Data Entry plug-in is invoked, which validates data entered by the user.
4. Depending on the provisioning approach, either asynchronous or synchronous provisioning procedures are invoked.

If provisioning is performed with the Provisioning Console, then after the Pre-Data Entry Plug-in is invoked, but before the Post-Data Entry plug-in is invoked, an administrator can modify the application attributes.

Overview of Provisioning Methodologies

This section describes the procedures for provisioning users in Oracle Identity Management. It contains these topics:

- Provisioning Users from the Provisioning Console
- Provisioning Users that are Synchronized from an External Source
- Provisioning Users Created with Command-Line LDAP Tools
- Bulk Provisioning
- On-Demand Provisioning
- Application Bootstrapping

Provisioning Users from the Provisioning Console

You can use the Provisioning Console to centrally manage user provisioning and deprovisioning of one or more users simultaneously. The console includes a wizard-based interface for creating, modifying, and deleting individual users, and for

selectively provision and deprovision users for any provisioning-integrated applications. The Provisioning Console also supports bulk user creation, modification, and deletion of users from an LDIF file. See "Bulk Provisioning" on page 12-8 for more information.

Provisioning Users that are Synchronized from an External Source

When Oracle Internet Directory is used as a central repository and enterprise user entries are synchronized from third-party directories to Oracle Internet Directory, each user identity is automatically provisioned according to the default provisioning policy of each provisioning-integrated application.

Provisioning Users Created with Command-Line LDAP Tools

Any tools developed by Oracle or third-party vendors that use standard command-line LDAP tools can create user entries in Oracle Internet Directory. As with user entries that are synchronized from external sources, any user entries created with command-line LDAP tools or any other means are provisioned according to the default provisioning policies for each provisioning-integrated application.

Bulk Provisioning

You can use the Provisioning Console or the Directory Integration and Provisioning Assistant to create and provision user entries by providing an LDIF (LDAP Data Interchange Format) file containing user data. The LDIF file should contain only LDAP-specific attributes. When user entries in an LDIF file are created in Oracle Internet Directory, each entry is provisioned according to the default provisioning policy of each provisioning-integrated application.

On-Demand Provisioning

On-demand provisioning occurs when a user attempts to access an application and the application has no knowledge of the user in its repository. The application determines whether to provision a user account based on its default provisioning policies. After provisioning a user account in its repository, an application will update the provisioning status of the user entry in Oracle Internet Directory.

Application Bootstrapping

The Oracle Provisioning Service notifies newly registered applications of all existing user entries in Oracle Internet Directory and attempts to provision each existing user entry as if they were a new user in the application.

Organization of User Profiles in Oracle Internet Directory

This section discusses the organization of user profiles in Oracle Internet Directory. It contains these topics:

- Organization of Provisioning Entries in the Directory Information Tree
- Understanding User Provisioning Statuses

Organization of Provisioning Entries in the Directory Information Tree

The Oracle Provisioning Service relies on user profiles in the DIT that consist of attributes containing personal information and preferences for the various applications

in which the user is provisioned. These user attributes for the Oracle Provisioning Service can be categorized as follows:

- Base attributes that are available for every user entry
- Application-specific attributes that are only available if a user is provisioned in an application

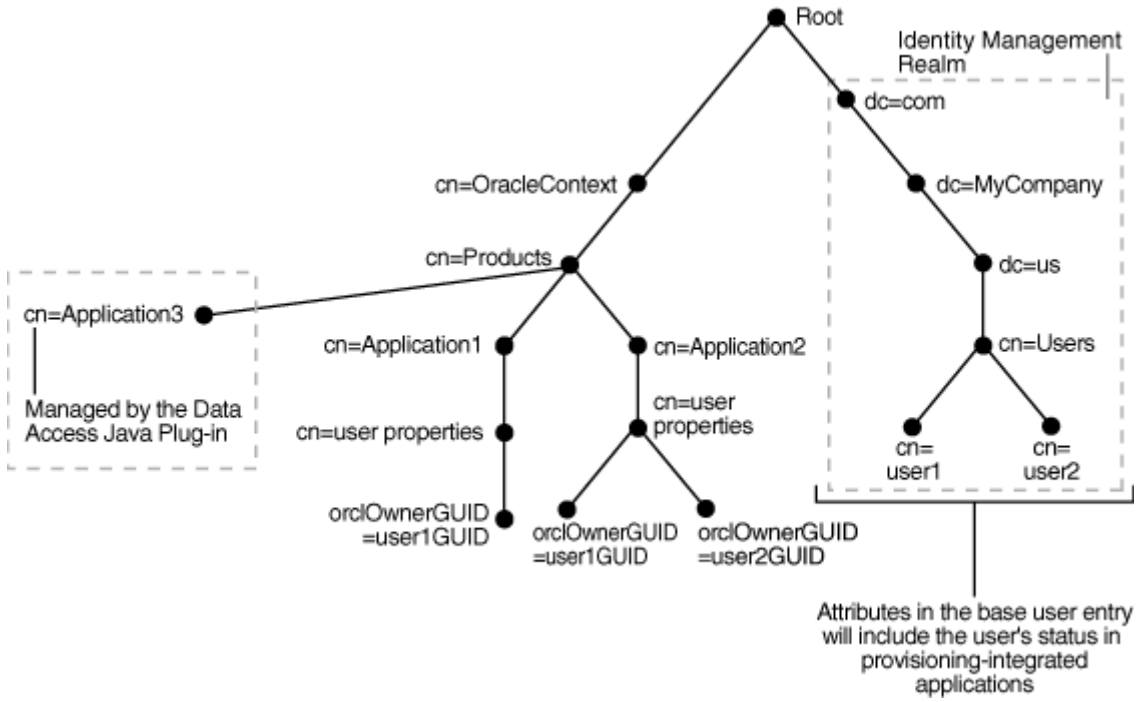
Base user attributes primarily belong to standard LDAP object classes such as `organizationalPerson` and `inetOrgPerson`, and consist of personal details that include first name, last name, given name, e-mail address, and telephone numbers. Base user attributes also consist of Oracle application-specific attributes that belong to the `orclUserV2` auxilliary class.

Oracle Internet Directory is the primary repository for both base attributes and application-specific attributes. Both types of attributes are stored in each user's profile. However, an application may cache user attributes that are updated with the provisioning event notification service.

As shown in Figure 12–6, user attributes are stored in two locations within the DIT. Base user entries, which include attributes belonging to `inetorgperson` and `orcluserv2`, are stored under `cn=users,`*`Realm DN`*. The provisioning status of each user entry is also stored in the base user entry. Application-specific attributes reside in separate entries in the application container. The LDAP schema relating to the application-specific attribute definitions and the object classes are created during the install or upgrade process. Application-specific attributes are qualified by an auxiliary object class, which will enable searching for the application-specific user properties of the entry. By default, application-specific entries are stored as `orclOwnerGUID=`*`GUID of the Base User`* under the `cn=User Properties, cn=`*`Application Type`*`, cn=Products,cn=OracleContext,` *`Realm DN`* container.

Some applications manage their own application attributes and implement the Data Access Java plug-in, which is described in "Understanding Provisioning Concepts" on page 12-2. The Oracle Provisioning Service invokes this plug-in whenever the base user attributes or application-specific attributes are modified.

Figure 12–6 Base User and Application-Specific Attributes



Understanding User Provisioning Statuses

This section discusses the user provisioning statuses in Oracle Internet Directory. It contains these topics:

- Provisioning Status in Oracle Internet Directory
- Provisioning Status Transitions
- Upgrading and Coexistence Provisioning Statuses
- Provisioning Statuses and Exception Handling

Provisioning Status in Oracle Internet Directory

The Oracle Provisioning Service records a user's provisioning status in Oracle Internet Directory for each provisioning-integrated application. Provisioning status can be set by the Oracle directory integration and provisioning server, with bulk provisioning using the Directory Integration and Provisioning Assistant, or by a provisioning-integrated application. Table 12–1 lists the provisioning statuses.

Table 12–1 Provisioning Statuses in Oracle Internet Directory

Internal Status	GUI Status	Description
Provisioning Statuses		
PROVISIONING_REQUIRED	Pending	Provisioning required. This status is selected by an administrator or set according to an application's provisioning policies. Note that this status does determine whether a user has been provisioned.
PROVISIONING_IN_PROGRESS	In Progress	Provisioning in progress. The user can access the application when this is the current status. if the application performs provisioning at scheduled intervals. The application may also provision the user on-demand.
PROVISIONING_SUCCESSFUL	Successful	Provisioning successful. This status is updated automatically by the Oracle directory integration and provisioning server, with bulk provisioning using the Directory Integration and Provisioning Assistant, or a provisioning-integrated application.
PROVISIONING_NOT_REQUIRED	Not Requested	Provisioning not required. This status is selected by an administrator or set according to an application's provisioning policies. Note that this status does determine whether a user will not be provisioned.
PROVISIONING_FAILURE	Failed	Provisioning failed. This status is updated automatically by the Oracle directory integration and provisioning server, with bulk provisioning using the Directory Integration and Provisioning Assistant, or a provisioning-integrated application. The user cannot access the application when this is the current status.
Deprovisioning Statuses		
DEPROVISIONING_REQUIRED	Pending de-provisioning	Deprovisioning required. The user is still provisioned when this is the current status.
DEPROVISIONING_IN_PROGRESS	De-provisioning In Progress	Deprovisioning in progress.
DEPROVISIONING_SUCCESSFUL	Successfully de-provisioned	Deprovisioning successful. The user cannot access the application when this is the current status.
DEPROVISIONING_FAILURE	Failed de-provisioning	Deprovisioning failed. The user is still provisioned when this is the current status.
Upgrade Statuses		
PENDING_UPGRADE	Pending Upgrade	Provisioning upgrade pending.
UPGRADE_IN_PROGRESS	Upgrade In Progress	Provisioning upgrade in progress.
UPGRADE_FAILURE	Upgrade Failed	Provisioning upgrade failed.

The provisioning status for each application is stored in the `orclUserApplnProvStatus` attribute in a user entry. This attribute is indexed in Oracle Internet Directory and searchable. A subtyped `orclUserApplnProvStatus` attribute is created for each provisioning-integrated application. For example, the following statements store a user's provisioning statuses for an e-mail application and a scheduling application. The user's provisioning status for the e-mail application is `PROVISIONING_SUCCESS` while his or her provisioning status for the scheduling application is `PROVISIONING_FAILURE`.

```
orclUserApplnProvStatus;CORP-MAIL_E-MAIL:PROVISIONING_SUCCESS
orclUserApplnProvStatus;CORP-SCHEDULE_CALENDAR:PROVISIONING_FAILURE
```

Additional information about a user's provisioning status in an application is stored in the `orclUserApplnProvStatusDesc` attribute and the provisioning failure account for each application is stored in the `orclUserApplnProvFailureCount` attribute. As with the `orclUserApplnProvStatus` attribute, separate `orclUserApplnProvStatusDesc` and `orclUserApplnProvFailureCount` attributes are created for each provisioning-integrated application. The format for the `orclUserApplnProvStatusDesc` attribute is the same as the `orclUserApplnProvStatus` attribute, except that a timestamp and descriptive information are appended to the application name and type, as follows:

```
orclUserApplnProvStatusDesc;CORP-MAIL_E-MAIL:20040101010101^Missing employee ID
```

The `orclUserApplnProvStatus`, `orclUserApplnProvStatusDesc`, and `orclUserApplnProvFailureCount` attributes are contained in the `orclUserProvStatus` object class as optional attributes.

Provisioning Status Transitions

Table 12–2 lists the valid provisioning status transitions.

Table 12–2 Valid Provisioning Status Transitions in Oracle Internet Directory

Internal Status	GUI Status	Valid Transition From
Provisioning Statuses		
`PROVISIONING_REQUIRED`	Pending	*Initial missing state* `DEPROVISIONING_SUCCESSFUL`
`PROVISIONING_IN_PROGRESS`	In Progress	`PROVISIONING_REQUIRED`

`PROVISIONING_SUCCESSFUL`	Successful	`PROVISIONING_REQUIRED`
		`PROVISIONING_IN_PROGRESS`
		`PROVISIONING_FAILURE`

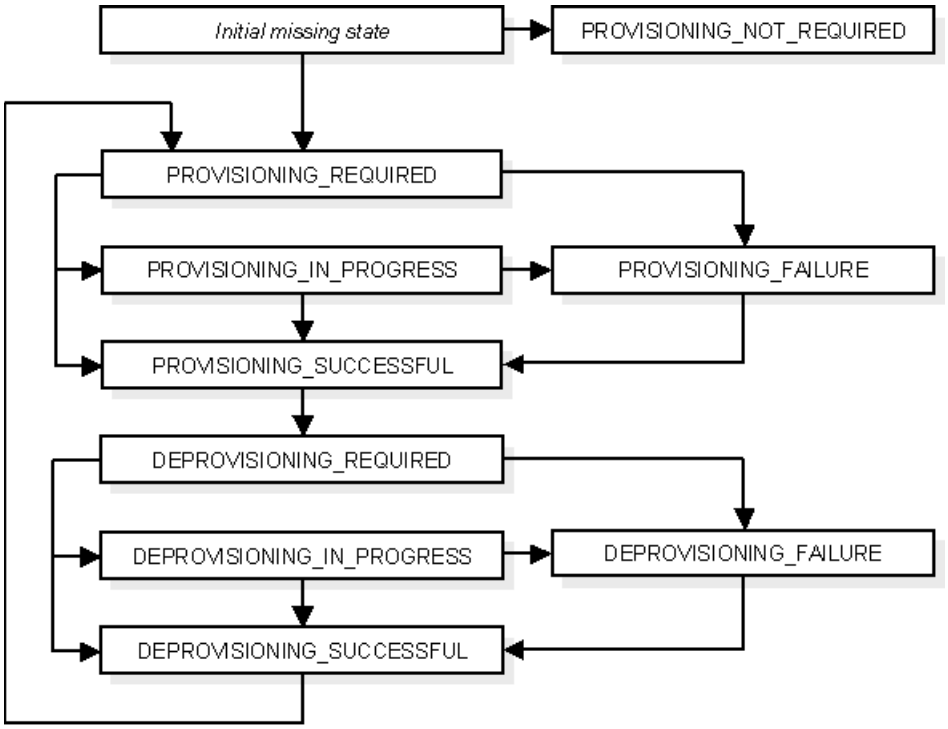
`PROVISIONING_NOT_REQUIRED`	Not Requested	*Initial missing state*
`PROVISIONING_FAILURE`	Failed	`PROVISIONING_REQUIRED` `PROVISIONING_IN_PROGRESS`
Deprovisioning Statuses		
`DEPROVISIONING_REQUIRED`	Pending de-provisioning	`PROVISIONING_SUCCESSFUL`

Table 12–2 (Cont.) Valid Provisioning Status Transitions in Oracle Internet Directory

Internal Status	GUI Status	Valid Transition From
DEPROVISIONING_IN_PROGRESS	De-provisioning In Progress	PROVISIONING_SUCCESSFUL
DEPROVISIONING_SUCCESSFUL	Successfully de-provisioned	DEPROVISIONING_REQUIRED
		DEPROVISIONING_IN_PROGRESS
		DEPROVISIONING_FAILURE
DEPROVISIONING_FAILURE	Failed de-provisioning	DEPROVISIONING_REQUIRED
		DEPROVISIONING_IN_PROGRESS

Figure 12–7 illustrates the valid provisioning status transitions.

Figure 12–7 Valid Provisioning Status Transitions



Upgrading and Coexistence Provisioning Statuses

In Oracle Identity Management 10*g* Release 2 (10.1.2), a user entry can be physically represented in Oracle Internet Directory by multiple LDAP entries. In addition to the base user entry, separate LDAP entries may exist for each provisioning-integrated application.

In a typical upgrade of Oracle Identity Management, multiple middle tiers are not upgraded simultaneously. This means that following an Oracle Identity Management upgrade, middle tiers from a previous version may need to run in parallel with middle tiers from the upgraded version. When a middle tier is upgraded, all of a user's application-specific data that was previously stored in the application metadata repository, will be migrated on-demand. For each user entry that is present in Oracle Internet Directory prior to the upgrade, the Oracle directory integration and provisioning server will initiate a new user event and assign a provisioning status of PENDING_UPGRADE to the user entry. If a new user entry is created from an older

middle tier or some unsupported route, such as an existing application using the standard LDAP SDK, the provisioning status attribute will be missing. In this case, the Oracle directory integration and provisioning server also initiates a new user event and assign a provisioning status of `PENDING_UPGRADE` to the user entry.

Once a provisioning-integrated application receives the event, it will return a response to the Oracle directory integration and provisioning server indicating whether the user is provisioned or not provisioned. The Oracle directory integration and provisioning server then updates the provisioning status in the user entry accordingly.

Provisioning Statuses and Exception Handling

If a new user entry created with the Provisioning Console or through synchronization with an external data source does not contain enough information to provision the user in a particular application, provisioning may fail. Provisioning can also fail for a variety of other reasons. The Oracle Provisioning Service identifies user provisioning failures as exceptions.

Whenever an application responds to a `USER_ADD` event with a failure status, the Oracle directory integration and provisioning server will change the user's provisioning status to `PROVISIONING_FAILURE`. The directory integration and provisioning server will then send notifications to the applications of the failed cases also just like a new user case. This will serve as a retry for the provisioning request.

The provisioning status of a user displays in the Provisioning Console. The administrator can make the necessary changes to fix the problem and the provisioning would get retried automatically. This will result in invocation of the data access plug in if the provisioning is synchronous. However, an event will be propagated if the provisioning is asynchronous.

This sequence of steps would be retried again as long as the user is not provisioned successfully.

Understanding Provisioning Flow

This section discusses the flow of information and control in various provisioning scenarios. It contains these topics:

- Creating/Modifying Users with the Provisioning Console
- Deleting Users with the Provisioning Console
- User Provisioning From an External Source

Creating/Modifying Users with the Provisioning Console

You can use the Provisioning Console to create and provision new user entries in Oracle Internet Directory. The console uses a wizard-based interface to perform the following steps:

1. The initial user creation screen shows a list of required base user attributes. The base user attributes are populated after the Provisioning Console invokes the Pre-Data Entry plug-in. For user creation, the plug-in processes the base user attributes and generates the application's default provisioning policy and attributes. For user modification, the Provisioning Console retrieves user information from Oracle Internet Directory and the plug-in retrieves application information.
2. The next step in the wizard displays how a user will be provisioned in each application, based on the application's default provisioning policy. For user

modification, this step displays one list with applications for which the user is currently provisioned and another list in which the user can be provisioned. You can select one of the following values for application in which the user is not yet provisioned:

- **User Policy**. The selected value for this field is based on each application's default provisioning policy. This field can display one of two values: Provision or Do Not Provision.
- **Override Policy to perform Provision**. Selecting this option overrides the application's default policy and provisions the user.
- **Override Policy NOT to perform Provision**. Selecting this option override the application's default policy and does not provision the user.

For applications in which the user is currently provisioned, there will be an option for deprovisioning the user.

3. For applications in which the user is not provisioned, the next step in the wizard displays attributes for the applications to be provisioned, with the default values returned by the Pre-Data Entry plug-in. For applications in which the user is provisioned, current application information is listed. You can make any necessary changes to the attributes in this step before clicking the Next button. When you click the Next button, the Post-Data Entry plug-in is invoked, which validates the data you entered.
4. The final step in the wizard enables you to review application attributes and values. After you click the Finish button, the Provisioning Console creates or updates the user information in Oracle Internet Directory, and then invokes the Data Access Java plug-in for applications that are provisioned synchronously to create or update the application

Deleting Users with the Provisioning Console

Before a user is deleted, the Provisioning Console displays a read-only page listing the base user and the application attributes. After the user confirms the deletion, the Provisioning Console deletes the base user information and any application-specific information or invokes the Data Access Java plug-in for applications that are provisioned synchronously. For asynchronous applications, a `USER_DELETE` event is propagated.

User Provisioning From an External Source

The majority of the deployments are expected to provision users from an external source, such a third-party enterprise user repository. In these types of deployments, the third-party repository bootstraps Oracle Internet Directory. Oracle Directory Integration and Provisioning will provide ongoing synchronization between Oracle Internet Directory and the third-party repository. Example of third-party user repositories include Oracle Human Resources and LDAP directories such as Microsoft Active Directory, and SunONE Directory Server.

The Oracle Directory Synchronization Service will create the user entry in Oracle Internet Directory. Since the information coming from the external source may not be sufficient to provision the user in various applications, the application defaults will be used to create the application information. User creation by the Oracle Directory Synchronization Service occurs as follows:

1. The Oracle Directory Synchronization Service evaluates the provisioning policies specified by the applications to determine whether the user should be provisioned in the application.
2. The Oracle Directory Synchronization Service evaluates any other plug-ins that the application has registered.
3. The Oracle Provisioning Service invokes the PL/SQL plug-in or the Data Access Java plug-in to deliver the user information to the application.
4. The provisioning status of the user is returned by the application using the event interfaces.
5. The Oracle Provisioning Service updates the provisioning status of the user for the application.

How are Administrative Privileges Delegated?

Administrative rights in Oracle Delegated Administration Services vary according to the privileges delegated to each administrator. An administrator may be granted rights to manage and provision users, manage applications, or any combination of these privileges, as described in the following scenarios:

- The Provisioning Administration Model
- Oracle Delegated Administration Services Privileges
- Provisioning Administration Privileges
- Application Administration Privileges
- Oracle Delegated Administration Services and Provisioning Administration Privileges
- Application Administration and Oracle Delegated Administration Services Privileges
- Provisioning and Application Administration Privileges
- Oracle Delegated Administration Services, Provisioning, and Application Administration Privileges

The Provisioning Administration Model

The following types of provisioning information is managed in Oracle Internet Directory:

- Base user information
- Application-specific information
- User provisioning status in each provisioning-integrated application; this information is stored in the base user entry but is administered separately

Administrators and users each require the following types of privileges:

- Administrators require privileges for managing base user attributes and application-specific information
- Users require privileges for managing their own base attributes and application-specific information

User accounts with administrative privileges are represented by the group entry "`cn=User Provisioning Admins,cn=Groups,cn=OracleContext`". In order to

manage application-specific information, the application must grant privileges to the "cn=User Provisioning Admins,cn=Groups,cn=OracleContext" group. If an application already defines a group with administrative privileges, then the application needs to add this group as a member of the group.

Oracle Delegated Administration Services Privileges

For administrators with privileges for Oracle Delegated Administration Services administration, Create, Delete, and Edit buttons are available in the Provisioning Console for performing user creation, deletion, and modification. When an administrator who only has administrative rights for Oracle Delegated Administration Services clicks one of these buttons, single-step procedures are used for performing the function.

Provisioning Administration Privileges

For administrators with provisioning privileges, Create, Delete, and Edit buttons are also available in the Provisioning Console for performing user creation, deletion, and modification. However, unlike the single-step procedures that occur for administrators with Oracle Delegated Administration Services privileges, wizard-based procedures perform creation and modification for administrators with provisioning privileges. User deletion is performed with the same single-step procedure that is available with Oracle Delegated Administration Services privileges, as described in "Oracle Delegated Administration Services Privileges" on page 12-17.

Application Administration Privileges

For administrators with application administration privileges, but not Oracle Delegated Administration Services privileges or provisioning privileges, Create and Delete buttons are not available in the Provisioning Console. However, an Edit button is available that launches the same wizard that is available with provisioning administration privileges, as described in "Provisioning Administration Privileges" on page 12-17. If the application administrator does not have provisioning privileges, then the first page in the wizard, which is used for general user provisioning, is read-only. Yet, the application administrator can modify the application provisioning attributes that are available on other pages in the wizard.

Oracle Delegated Administration Services and Provisioning Administration Privileges

Administrators with Oracle Delegated Administration Services privileges and provisioning privileges have the same rights that are available with provisioning administration privileges, as described in "Provisioning Administration Privileges" on page 12-17.

Application Administration and Oracle Delegated Administration Services Privileges

This section explains how privileges are delegated if an administrator is assigned various Oracle Delegated Administration Services privileges and also has administrative privileges.

Application Administration Privileges and Oracle Delegated Administration Services User Creation Privileges

For application administrators with user creation privileges in Oracle Delegated Administration Services, but not user editing or deletion privileges, the Create and Edit buttons are available in the Provisioning Console, but not the Delete button. User

creation is performed with the same wizard-based procedure that is available with provisioning administration privileges, as described in "Provisioning Administration Privileges" on page 12-17. User editing privileges are the same as those available with application administration privileges, as described in "Application Administration Privileges" on page 12-17.

Application Administration Privileges and Oracle Delegated Administration Services User Editing Privileges

For application administrators with user editing privileges in Oracle Delegated Administration Services, but not user creation or deletion privileges, the Edit button is available in the Provisioning Console, but not the Create or Delete buttons. User editing is performed with the same wizard-based procedure that is available with provisioning administration privileges, as described in "Provisioning Administration Privileges" on page 12-17.

Application Administration Privileges and Oracle Delegated Administration Services User Deletion Privileges

For application administrators with user deletion privileges in Oracle Delegated Administration Services, but not user creation or modification privileges, the Delete and Edit buttons are available in the Provisioning Console, but not the Create button. User deletion is performed with the same single-step procedure that is available with Oracle Delegated Administration Services privileges, as described in "Oracle Delegated Administration Services Privileges" on page 12-17. User editing is performed with the same wizard-based procedure that is available with provisioning administration privileges, as described in "Provisioning Administration Privileges" on page 12-17.

Provisioning and Application Administration Privileges

Administrators with provisioning privileges and application administration privileges have the same rights that are available with provisioning administration privileges, as described in "Provisioning Administration Privileges" on page 12-17.

Oracle Delegated Administration Services, Provisioning, and Application Administration Privileges

Administrators with Oracle Delegated Administration Services privileges and application administration privileges have the same rights that are available with provisioning administration privileges, as described in "Application Administration Privileges" on page 12-17.

13

Deploying Provisioning-Integrated Applications

This chapter explains how to deploy provisioning-integrated applications with the Oracle Provisioning Service. It contains these topics:

- Deployment Overview for Provisioning-Integrated Applications
- Registering Applications for Provisioning
- Configuring Application Provisioning Properties

See Also:

- Chapter 4, "Managing the Oracle Directory Integration and Provisioning Server"
- "Troubleshooting Provisioning" on page C-14

Deployment Overview for Provisioning-Integrated Applications

To deploy provisioning-integrated applications with the Oracle Provisioning Service, you perform these general steps:

1. Install Oracle Internet Directory, which includes Oracle Directory Integration and Provisioning .
2. Load user information into Oracle Internet Directory.

   See Also: *Oracle Internet Directory Administrator's Guide*

3. Start the Oracle directory integration and provisioning server by following the procedures in "Starting, Stopping, and Restarting the Oracle Directory Integration and Provisioning Server" on page 4-8.
4. Install the applications and use the Provisioning Subscription Tool to create a provisioning profile for each application.

   See Also: "The Provisioning Subscription Tool (oidprovtool)" on page 3-8

5. Configure application registration by following the procedures described in "Registering Applications for Provisioning" on page 13-2.

6. Configure application provisioning by following the procedures described in "Configuring Application Provisioning Properties" on page 13-4.
7. Periodically monitor the status of the provisioning event propagation for each application. You can do this by using the Oracle Enterprise Manager 10*g* Application Server Control Console.

> **See Also:** The chapter on logging, auditing, and monitoring the directory in *Oracle Internet Directory Administrator's Guide*

Registering Applications for Provisioning

After you install an application and use the Provisioning Subscription Tool to create a provisioning profile for it, you must perform the following steps to register the application for provisioning:

1. Perform the initial provisioning registration and create a provisioning-integration profile. The Oracle Provisioning Service uses the provisioning-integration profiles to identify provisioning-integrated applications.
2. Provide the Oracle Provisioning Service with application- specific attributes, default values, and whether an attribute is mandatory when provisioning users for the application.
3. Register any plug-ins that are required by the provisioning-integrated application. This may include application-specific plug-ins that the application uses to enforce business policies.

> **Note:** The Oracle Provisioning Service does not support instance-level provision of applications that support multiple instance architecture. If you install multiple instances of the same application, the Oracle Provisioning Service treats each instance as a separate provisioning-integrated application.

When creating users with the Provisioning Console, an administrator can assign user attributes for a specific provisioning-integrated application. Because Oracle Internet Directory is the primary directory for attributes that the Provisioning Console manages, application-specific attributes are stored in Oracle Internet Directory for each user that is provisioned for an application. For better performance, provisioning-integrated applications usually cache a local copy of user attributes instead of retrieving them from Oracle Internet Directory. Applications are notified of user creations, user deletions, and attribute modifications either synchronously with the Data Access Java plug-in or asynchronously with a PL/SQL plug-in.

Registration creates a unique identify for an application in Oracle Internet Directory. Oracle applications typically register themselves for provisioning by using the repository APIs located in the repository.jar file, which Oracle Application Server installs by default in the $*ORACLE_HOME*/jlib directory. In addition to creating an application entry in Oracle Internet Directory, the repository APIs can be used to add applications to privileged groups.

For non-Oracle applications that are not capable of using the registration APIs, you can use LDAP commands and LDIF templates to create identities for the applications in Oracle Internet Directory. You create a container for the application under `cn=Products,cn=OracleContext"` or `cn=Products, cn=OracleContext, Realm DN`. The container where you create an application identity depends on whether the application will be available to users in a single realm or multiple realms.

In most case, you should create an application identity in the `cn=Products, cn=OracleContext` container so the application is not bound by the identity management policies of a specific Oracle Internet Directory identity management realm.

You can install multiple instances of the same application. Installing a new instance of a provisioning-integrated application creates a separate entry for the new instance under the application identity container. Although some configuration settings are instance-specific, other settings are shared across multiple instances of the same application. As an example, consider an application that is similar to Oracle Files. You can deploy multiple instances of Oracle Files in an environment where each instance is independent of other instances. You define each instance as a separate provisioning-integrated application. You can also provision users in multiple instances of the application.

When you install the first instance of an application, you must create in Oracle Internet Directorythe entries shown in the following example. The example creates the application identity in the `cn=Products, cn=OracleContext` container and assumes the application name and type are Files-App1 and FILES.

```
dn: cn=FILES,cn=Products,cn=OracleContext
changetype: add
objectclass: orclContainer

dn: orclApplicationCommonName=Files-App1,cn=FILES,cn=Products,cn=OracleContext
changetype: add
orclappfullname: Files Application Instance 1
userpassword: password
description: This is a test application instance.
protocolInformation: protocol information
orclVersion: 1.0
orclaci: access to entry by group="cn=odisgroup,cn=DIPAdmins,cn=Directory
Integration Platform,cn=Products,cn=OracleContext" (browse,proxy) by
group="cn=User Provisioning Admins,cn=Groups,cn=OracleContext" (browse,proxy)
orclaci: access to attr=(*) by group="cn=odisgroup,cn=DIPAdmins,cn=Directory
Integration Platform,cn=Products,cn=OracleContext" (search,read,write,compare) by
group="cn=User Provisioning Admins,cn=Groups,cn=OracleContext"
(search,read,write,compare)
```

When you install the second instance of an application, you must create in Oracle Internet Directorythe entries shown in the following example. The example also creates the application identity in the `cn=Products, cn=OracleContext` container and assumes the application name is Files-App2.

```
dn: orclApplicationCommonName=Files-App2,cn=FILES,cn=Products,cn=OracleContext
changetype: add
orclappfullname: Files Application Instance 2
userpassword: password
description: This is a test Appliction instance.
protocolInformation: protocol information
orclVersion: 1.0
orclaci: access to entry by group="cn=odisgroup,cn=DIPAdmins,cn=Directory
Integration Platform,cn=Products,cn=OracleContext" (browse,proxy) by
group="cn=User Provisioning Admins,cn=Groups,cn=OracleContext" (browse,proxy)
orclaci: access to attr=(*) by group="cn=odisgroup,cn=DIPAdmins,cn=Directory
Integration Platform,cn=Products,cn=OracleContext" (search,read,write,compare) by
group="cn=User Provisioning Admins,cn=Groups,cn=OracleContext"
(search,read,write,compare)
```

After you successfully register a provisioned-integrated application with Oracle Internet Directory, you may need to add the application to various privileged groups. Table 13–1 lists common privileged groups in Oracle Internet Directory.

Table 13–1 Common Privileged Groups in Oracle Internet Directory

Group	Description
`OracleDASCreateUser`	Create users
`OracleDASEditUser`	Edit users
`OracleDASDeleteUser`	Delete users
`OracleDASCreateGroup`	Create groups
`OracleDASEditGroup`	Edit groups
`OracleDASDeleteGroup`	Delete groups

The following LDIF file demonstrates how to grant create user privileges in all realms to the Files-App1 application:

```
dn:cn=OracleCreateUser,cn=Groups,cn=OracleContext
changetype: modify
add: uniquemember
uniquemember:
orclApplicationCommonName=Files-App1,cn=FILES,cn=Products,cn=OracleContext
```

Configuring Application Provisioning Properties

After you register a provisioning-integrated application, you must configure its properties. Each application's provisioning profile maintains its own provisioning configuration properties. Provisioning-integrated applications use properties to store the following types of metadata:

- Application identity information
- Identity realm information
- Default application provisioning policies
- Application attribute properties and defaults
- Application provisioning plug ins
- Application event interface information
- Application event propagation information

The Oracle Provisioning Service supports three versions of provisioning profiles: 1.1, 2.0, and 3.0. Version 3.0 provisioning profiles are only available with Oracle Identity Management 10*g* Release 2 (10.1.2). Different applications support different provisioning profile versions. For example, many Oracle applications only support version 2.0. However, Oracle Collaboration Suite supports provisioning profile version 3.0. The primary differences between the provisioning profile versions are as follows:

- You can only use the Provisioning Console to provision target applications that support provisioning profile version 3.0. Although applications that only support provisioning profile versions 1.1 and 2.0 will not be available in the Provisioning Console, they will be notified of events for which they are configured.
- Provisioning applications that support provisioning profile versions 1.1 and 2.0 is a single-step process involving the `oidprovtool` utility, which is described in

"The Provisioning Subscription Tool (oidprovtool)" on page 3-8. However, provisioning applications that support provisioning profile version 3.0 is a multiple step process, which is described in centralized user provisioning Java API reference chapter of the *Oracle Identity Management Application Developer's Guide*.

- The Oracle Provisioning Service only maintains user provisioning status for applications that support provisioning profile version 3.0.

See Also: The centralized user provisioning Java API reference chapter of the *Oracle Identity Management Application Developer's Guide*

14

Managing with the Provisioning Console

This chapter explains how to manage with the Oracle Internet Directory Provisioning Console. It contains these topics:

- Managing Users with the Provisioning Console
- Managing Applications with the Provisioning Console

See Also:

- Chapter 4, "Managing the Oracle Directory Integration and Provisioning Server"
- "Troubleshooting Provisioning" on page C-14

Managing Users with the Provisioning Console

This section describes how to manage users with the Provisioning Console. It contains these topics:

- Searching for Users Based on Provisioning Criteria
- Creating Users with the Provisioning Console
- Provisioning and De-Provisioning Users with the Provisioning Console

Note: User administration that is not specifically related to provisioning, such as user deletion, is handled by the Oracle Internet Directory Self-Service Console. For more information, see the *Oracle Identity Management Guide to Delegated Administration*.

Searching for Users Based on Provisioning Criteria

To search for users based on provisioning criteria:

1. Select the **Directory** tab, then select **Users**. From the Users page, click **Provisioning Search**. The Provisioning Search window appears.

   This window is described in *Oracle Identity Management Guide to Delegated Administration*.

2. Select one of the following options to determine how you want to search for users:
   - **Show users that match all conditions**
   - **Show users that match any condition**

3. Select one of the following conditions from the first box to the right of each application that you want to search for a user's provisioning status:
   - is (default)
   - is not
   - is present
   - is not present
4. Select one of the following provisioning statuses from the second box to the right of each application that you want to search.
   - Pending
   - Not Requested
   - Successful
   - Failed
   - In Progress
   - Pending de-provisioning
   - Successfully de-provisioned
   - Failed de-provisioning
   - De-provisioning In Progress
   - Pending Upgrade
   - Upgrade in Progress
   - Upgrade Failed
5. To add additional search attributes, select an attribute name from the **Add Another** box, then click **Add**.
6. Choose **Go** to display the entries that match the criteria you entered.

Creating Users with the Provisioning Console

To create a user with the Provisioning Console:

1. In the Oracle Internet Directory Self-Service Console, select the **Directory** tab, then select **Users**. The Search for Users window appears.

   This window is described in *Oracle Identity Management Guide to Delegated Administration*.
2. Select **Create** to display the General Provisioning window.

   This window is described in *Oracle Identity Management Guide to Delegated Administration*.
3. In the General Provisioning window, enter the appropriate information. To reset the password for an existing user entry, enter a new value in the Password field.

   Caution: The User ID field cannot contain spaces or any of the following characters: & ' % ? \ / + = ( ) * ^ , ; | ' ~
4. Choose **Next** to display the Application Provisioning window.

This window is described in *Oracle Identity Management Guide to Delegated Administration*.

5. In the Application Provisioning window, select the applications for which you want to provision the user entry. The available applications listed in this window will vary according to your environment. The default provisioning policy determines which applications are provisioned by default whenever a new user is created. Depending on the default policy, you may be able to override the policy for one or more applications. If policy override is not available, the Provision or Do Not Provision columns (depending on the default) will be grayed out.

   To change the default provisioning policy for an application, follow the instructions in "Managing Applications with the Provisioning Console" on page 14-5.

   Note: In Oracle Application Server 10*g* Release 2 (10.1.2), only components that are part of Oracle Collaboration Suite can be provisioned with the Provisioning Console.

6. Choose **Next** to display the Application Attributes window.

   This window is described in *Oracle Identity Management Guide to Delegated Administration*.

7. In the Application Attributes window, enter attribute values for the applications you selected to provision for the user entry. Depending on how your applications are configured, default values may be entered for some of the attributes.

   See Also: "Managing Applications with the Provisioning Console" on page 14-5

8. Choose **Next** to display the Provisioning Review window.

   This window is described in *Oracle Identity Management Guide to Delegated Administration*.

9. After reviewing the provisioning options for the user entry, choose **Finish**.

Provisioning and De-Provisioning Users with the Provisioning Console

To provision or de-provision a user with the Provisioning Console:

1. In the Oracle Internet Directory Self-Service Console, select the **Directory** tab, then select **Users**. The Search for Users window appears.

   This window is described in *Oracle Identity Management Guide to Delegated Administration*.

2. In the **Search for User** field, enter the first few characters of the user's first name, last name, e-mail address, or user ID. For example, if you are searching for Anne Smith, you could enter `Ann` or `Smi`. To generate a list of all users in the directory, leave this field blank.

3. Choose **Go** to display the search results.

4. Select the user you want to provision or de-provision, then choose **Edit** to display the General Provisioning window.

This window is described in *Oracle Identity Management Guide to Delegated Administration*.

> **Note:** If you do not have sufficient privileges to edit a user entry, then the **Edit** button does not appear.

5. In the General Provisioning window, enter the appropriate information. To reset the password for an existing user entry, enter a new value in the Password field.

> **Caution:** The User ID field cannot contain spaces or any of the following characters: & ' % ? \ / + = ( ) * ^ , ; | ' ~

6. Choose **Next** to display the Application Provisioning window.

   This window is described in *Oracle Identity Management Guide to Delegated Administration*.

7. In the Application Provisioning window, select the applications for which you want to provision or de-provision the user entry. The available applications listed in this window will vary according to your environment. The default provisioning policy determines which applications are provisioned by default whenever a new user is created. Depending on how your applications are configured, you may be able to override the policy for one or more applications. If policy override is not available, the Provision or Do Not Provision columns (depending on the default) will be grayed out.

   To change the default provisioning policy for an application, follow the instructions in "Managing Applications with the Provisioning Console" on page 14-5.

> **Note:** In Oracle Application Server 10*g* Release 2 (10.1.2), only components that are part of Oracle Collaboration Suite can be provisioned with the Provisioning Console.

8. Choose **Next** to display the Application Attributes window.

   This window is described in *Oracle Identity Management Guide to Delegated Administration*.

9. In the Application Attributes window, enter attribute values for the applications you selected to provision for the user entry. Depending on your environment, default values may be entered for some of the attributes

> **See Also:** "Managing Applications with the Provisioning Console" on page 14-5

10. Choose **Next** to display the Provisioning Review window.

    This window is described in *Oracle Identity Management Guide to Delegated Administration*.

11. After reviewing the provisioning options for the user entry, choose **Finish**.

Managing Applications with the Provisioning Console

This section describes how to manage applications with the Provisioning Console. It contains these topics:

- Managing Application Defaults
- Reloading the Application Cache

Managing Application Defaults

This section explains how to manage defaults for provisioning-integrated applications. The available provisioning-enabled applications will vary according to your environment

> **Note:** In Oracle Application Server 10*g* Release 2 (10.1.2), only components that are part of Oracle Collaboration Suite can be provisioned with the Provisioning Console.

To manage application defaults:

1. Select the **Directory** tab, then select **Applications** to display the Manage Defaults: Select Application window.

   This window is described in *Oracle Identity Management Guide to Delegated Administration*.

2. In the Manage Defaults: Select Application window, select the applications for which you want to manage defaults.

3. Choose **Manage** to display the Manage Defaults: Attributes window.

   This window is described in *Oracle Identity Management Guide to Delegated Administration*.

4. In the Manage Defaults: Attributes window, enter default values in the attribute fields for the applications you selected in the Manage Defaults: Select Application window.

5. Choose **Submit**.

Reloading the Application Cache

The application cache determines the provisioning-integrated applications that are available in the Provisioning Console. You should reload the application cache whenever a provisioning-integrated application is enabled or disabled in Oracle Internet Directory.

To reload the application cache:

1. In the Provisioning Console, select the **Directory** tab, then select **Applications**. The Manage Defaults: Select Application window appears.

   This window is described in *Oracle Identity Management Guide to Delegated Administration*.

2. In the **Manage Defaults: Select Application** window, click the **Refresh** button.

15

Understanding the Oracle Provisioning Event Engine

This chapter discusses the Oracle provisioning event engine. It contains these topics:

- What are the Oracle Provisioning Events?
- Working with the Oracle Provisioning Event Engine

What are the Oracle Provisioning Events?

The Oracle provisioning event engine sends `USER_ADD`, `USER_MODIFY` and `USER_DELETE` events, depending the operation performed on the user entries in Oracle Internet Directory. Since the user will be represented by multiple entries containing base user and application specific user information, applications can subscribe to all of the attributes in the event.

The user events are also sent when a base entry or application entry is updated. However, no events are sent when an application entry is deleted because when an administrator requests the deprovisioning of a user from an application, a `USER_MODIFY` event is sent to the application with a provisioning status of `DEPROVISIONING_REQUIRED`. Once the application acknowledges the event by returning a value of `SUCCESS`, the application entry is deleted by the Oracle directory integration and provisioning server.

In order to receive notification of provisioning status changes, an application must subscribe to the `orclUserApplnProvStatus;`*`Application_Name`* attribute. For example to subscribe to the provisioning status change to an application named CORP_EMAIL, an application must subscribe to the `orclUserApplnProvStatus;CORP-EMAIL` attribute.

Working with the Oracle Provisioning Event Engine

The Oracle provisioning event engine generates events from add, modify, and delete operations that are performed on well-defined objects in Oracle Internet Directory. The Oracle provisioning event engine uses object definitions and event generation rules to generate events. This event generation model is very extensible because it enables you to define custom objects and event generation rules. The Oracle provisioning event engine object definitions and event generation rules are discussed in the following topics:

- Creating Custom Event Object Definitions
- Defining Custom Event Generation Rules

Creating Custom Event Object Definitions

Table 15–1 lists the properties that you can use to identify objects for which events can be generated.

Table 15–1 Event Object Properties

Property	Description
`ObjectName`	Assigns a unique name to identify the object
`ObjectCriteria`	Identifies the LDAP object class to use for identifying the object
`MustAttributeCriteria`	Provides any additional attributes that are required for identifying the object
`OptionalAttributeCriteria`	Provides any optional attributes that may be required for identifying the object
`FilterAttributeCriteria`	Lists the attributes that should not be sent during event propagation

Table 15–2 lists the predefined objects for which the Oracle provisioning event engine can generate events.

Table 15–2 Predefined Event Objects

Object Name	Valid Object Class Values
`Entry`	`*`
`User`	`orclUserV2,inetorgperson`
`Identity`	`orclUserV2,inetOrgPerson`
`Group`	`groupOfUniqueNames,orclGroup, orclPrivilegeGroup,groupOfNames`
`Subscription`	`orclServiceSubscriptionDetail`
`Subscriber`	`orclSubscriber`

Note: The metadata for event objects is stored in the following container: `cn=Object Definitions, cn=Directory Integration Platform,cn=Products,cn=OracleContext`

Defining Custom Event Generation Rules

You specify event generation rules in XML format. The DTD for event generation rules is as follows:

```
<?xml version='1.0' ?>
  <!DOCTYPE EventRuleSet [
    <!ELEMENT ChangeType (#PCDATA)>
    <!ELEMENT Rule  (#PCDATA)>
    <!ELEMENT EventName  (#PCDATA)>
    <!ELEMENT ResEvent (Rule*, EventName)>
    <!ELEMENT EventRule (ChangeType, ResEvent*)>
    <!ELEMENT EventRuleSet (EventRule*) >
  ]>
```

The element definitions in the preceding DTD are as follows:

- The `EventRuleSet` root element identifies a set of event rules for an individual event object.
- The `EventRuleSet` root element contains a list of `EventRule` elements.
- Each `EventRule` element depends on the value assigned to the `ChangeType` element.
- The `ChangeType` and `Rule` elements determine the event name to be propagated to an application.

Table 15–3 lists the event definitions that are supported by the Oracle provisioning event engine.

Table 15–3 Supported Event Definitions

Object Name	Change Type	Rule	Event Name
USER	Add	OrclApplnUserProvStatus=PENDING_UPGRADE	USER_ADD
	Add	OrclApplnUserProvStatus=PROVISIONING_REQUIRED	USER_ADD
	Modify	OrclApplnUserProvStatus= PENDING_UPGRADE	USER_ADD
		OrclApplnUserProvStatus=PROVISIONING_REQUIRED	USER_ADD
		OrclApplnUserProvStatus=PROVISIONING_FAILURE	USER_ADD
		OrclApplnUserProvStatus=DEPROVISIONING_REQUIRED	USER_MODIFY
		OrclApplnUserProvStatus=PROVISIONING_IN_PROGRESS	USER_MODIFY
		OrclApplnUserProvStatus=PROVISIONING_SUCCESSFUL	USER_MODIFY
	Delete	OrclApplnUserProvStatus=PROVISIONING_IN_PROGRESS	USER_DELETE
		OrclApplnUserProvStatus=PROVISIONING_SUCCESSFUL	USER_DELETE
		OrclApplnUserProvStatus=DEPROVISIONING_REQUIRED	
GROUP	Add		GROUP_ADD
	Modify		GROUP_MODIFY
	Delete		GROUP_DELETE
IDENTITY	Add		IDENTITY_ADD
	Modify		IDENTITY_MODIFY
	Delete		IDENTITY_DELETE
ENTRY	Add		ENTRY_ADD
	Modify		ENTRY_MODIFY
	Delete		ENTRY_DELETE
SUBSCRIPTION	Add		SUBSCRIPTION_ADD
	Modify		SUBSCRIPTION_ MODIFY
	Delete		SUBSCRIPTION_ DELETE

Table 15–3 (Cont.) Supported Event Definitions

Object Name	Change Type	Rule	Event Name
SUBSCRIBER	Add		SUBSCRIBER_ADD
	Modify		SUBSCRIBER_MODIFY
	Delete		SUBSCRIBER_DELETE

Note: The metadata for supported events objects is stored in the following container: `cn=Event Definitions, cn=Directory Integration Platform,cn=Products,cn=OracleContext`

16

Integration of Provisioning Data with the Oracle E-Business Suite

In Oracle Internet Directory 10*g* Release 2 (10.1.2), you can use the Oracle Provisioning Service to synchronize user accounts and other user information from the Oracle E-Business Suite.

> **See Also:** Oracle E-Business Suite documentation for further details on this integration and how to administer it
>
> The following notes on Oracle MetaLink at `http://metalink.oracle.com/`:
>
> - 233436.1—Installing Oracle Application Server 10*g* with Oracle E-Business Suite Release 11*i*
> - 261914.1—Integrating Oracle E-Business Suite Release 11*i* with Oracle Internet Directory and Oracle Application Server Single Sign-On
> - 233436.1—Oracle Application Server with Oracle E-Business Suite Release 11*i* Frequently Asked Questions

Part V

Integrating with Third-Party Identity Management Systems

This part discusses the concepts, components, and procedures involved in integrating with various third-party identity management systems. It contains these chapters:

17

Considerations for Integrating with Third-Party Directories

This chapter discusses the decisions you need to make before integrating with a third-party directory.

> **Note:** This chapter assumes that you are familiar with:
>
> - The chapter in *Oracle Internet Directory Administrator's Guide* about the deployment of identity management realms
> - *Oracle Identity Management Guide to Delegated Administration*

This chapter contains these topics:

Preliminary Considerations for Integrating with a Third-Party Directory

If you are deploying Oracle Internet Directory in an enterprise that already has an LDAP directory server, then you must configure both directories to co-exist in the same environment.

The co-existence of directories can require either of two different types of deployments:

- Simple synchronization with Oracle Internet Directory to support Enterprise User Security. Use this approach if your environment supports enterprise users by using a database server.

To configure simple synchronization with Microsoft Active Directory, see Chapter 18, "Integration with the Microsoft Active Directory Environment".

To configure simple synchronization with SunONE Directory Server, see Chapter 20, "Integration with SunONE (iPlanet) Directory Server".

- Complete integration with the Oracle Application Server infrastructure. This enables all enterprise users to use the various components in the Oracle Application Server suite. Use this approach if your environment uses a third-party directory as the enterprise directory and deploys an Oracle Application Server suite of applications.

  Because all Oracle Application Server components depend on the identity management realm, complete integration with the Oracle Application Server infrastructure requires you to make some decisions about the container for that realm. Once you have made these decisions, you can configure bootstrapping and synchronization between the directories.

Choose Which Directory Is to Be the Central Enterprise Directory

The central enterprise directory is the source of truth for all user, group, and realm information in the enterprise. It can be either Oracle Internet Directory or a third-party directory.

This section contains these topics:

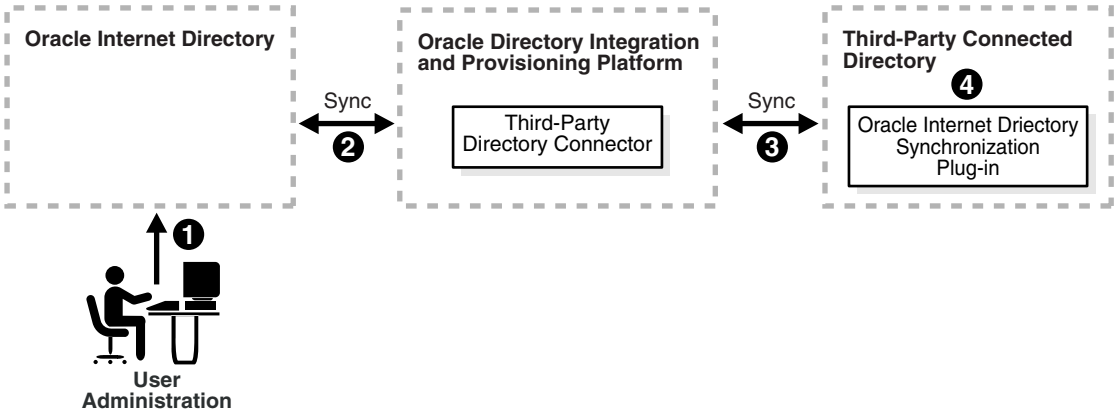
- Oracle Internet Directory as the Central Enterprise Directory
- Third-Party Directory as the Central Directory

Oracle Internet Directory as the Central Enterprise Directory

If Oracle Internet Directory is the central directory, then, once user, group, and realm objects are created, Oracle Internet Directory becomes the source of provisioning information for all Oracle components and third-party directories. The user and group objects for the entire enterprise are then provisioned in various Oracle components and third-party directories from Oracle Internet Directory.

Figure 17–1 shows a typical deployment in which Oracle Internet Directory is the central enterprise directory.

Figure 17–1 Interaction Between Components with Oracle Internet Directory as the Central Directory



As Figure 17–1 on page 17-2 shows, when Oracle Internet Directory is the central enterprise directory, typical provisioning of a user or group follows this process:

1. The user or group entry is created in Oracle Internet Directory by using the Oracle Internet Directory Self-Service Console, Oracle Directory Manager, or the command-line tools.
2. At the next scheduled interval, that entry creation event is read by the third-party directory connector in Directory Integration and Provisioning.
3. Following the mapping information in the integration profile, the user or group attributes in Oracle Internet Directory are appropriately mapped to the corresponding user or group attributes as required by the schema in the third-party directory.
4. The user and group entry is created in the third-party directory.

A user entry is modified in Oracle Internet Directory, when:

- A new attribute gets added to the entry
- The value of an existing attribute is modified
- An existing attribute is deleted

When Oracle Internet Directory is the central enterprise directory, the sequence of events during modification of a user or group entry is as follows:

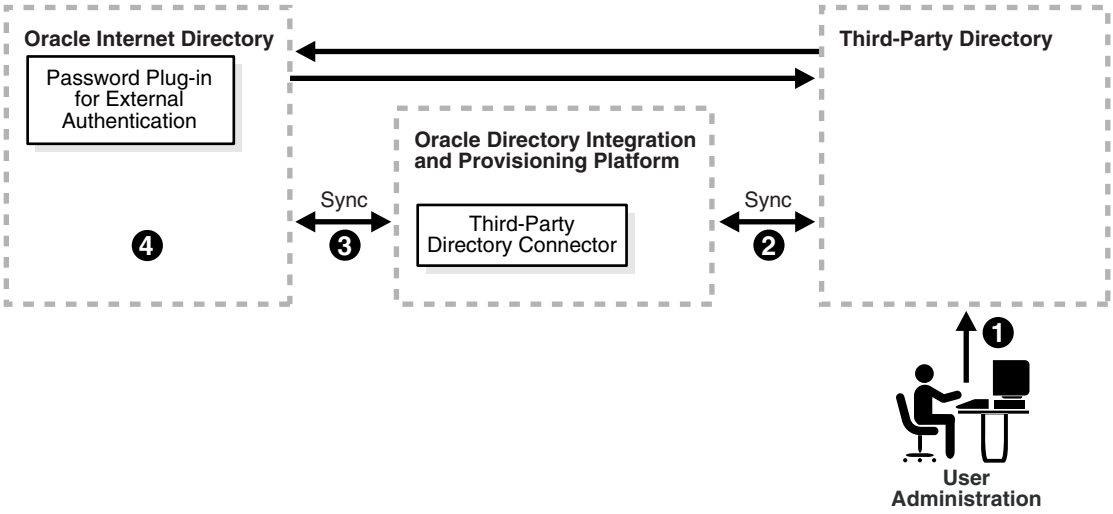
1. The entry is modified by using the Oracle Internet Directory Self-Service Console, Oracle Directory Manager, or the command-line tools.
2. At the next scheduled interval, that entry modification event is read by the third-party directory connector in Directory Integration and Provisioning,
3. Following the mapping information in the integration profile, the attribute in Oracle Internet Directory is appropriately mapped to the corresponding attribute in the connected directory
4. The user entry is modified in the third-party directory.

Third-Party Directory as the Central Directory

If a third-party directory is the central directory, then, once user, group, and realm objects are created, the third-party directory becomes the source of provisioning information for all Oracle components and other directories. In this case, Oracle Internet Directory is deployed to support Oracle components. To provide this support, Oracle Internet Directory stores a footprint that enables it to identify entries in the third-party directory.

Figure 17–2 shows a typical deployment where a third-party directory is the central enterprise directory.

Figure 17–2 Interaction of Components with a Third-Party Directory as the Central Directory



Process for Provisioning of a User or Group

As Figure 17–2 shows, when a third-party directory is the central enterprise directory, typical provisioning of a user or group follows this process:

1. The user or group entry is created in the third-party directory.
2. At the next scheduled interval, the entry creation event is read by the third-party directory connector in Directory Integration and Provisioning.
3. Following the mapping information in the integration profile, the user or group attributes in the third-party directory are mapped to the corresponding attributes in Oracle Internet Directory.
4. The user or group entry is created in Oracle Internet Directory.

Process for Modifying a User or Group Entry

An entry is modified in the third-party directory when:

- A new attribute gets added to the entry
- The value of an existing attribute is modified
- An existing attribute is deleted

When a third-party directory is the central enterprise directory, modification of a user or group entry follows this process:

1. The entry is modified in the third-party directory.
2. At the next scheduled interval, that entry modification event is read by the third-party directory connector in Directory Integration and Provisioning,
3. Following the mapping information in the integration profile, the attribute in the third-party directory is appropriately mapped to the corresponding attribute in Oracle Internet Directory.
4. The user or group entry is modified in Oracle Internet Directory.

As Figure 17–2 shows, when a third-party directory is the central enterprise directory, modification of passwords happens asynchronously in the directory that serves as the password repository. This happens by using plug-ins.

Choose Where to Store Passwords

Regardless of which directory is the central enterprise directory, the password can be stored in one or both directories. There are advantages and disadvantages to each option. This section compares the two options, and contains these topics:

Advantages and Disadvantages of Storing the Password in One Directory

By reducing to one the number of points of possible attack, storing the password in only one directory can make the password more secure. Moreover, it eliminates synchronization issues when the password is modified.

On the other hand, storing the password in one directory provides a single point of failure for the entire network. If the directory that fails is a third-party one, then even though user footprints are available in Oracle Internet Directory, users cannot access Oracle components.

Moreover, although storing passwords in only the central directory eliminates any possible synchronization issues, it requires you to enable applications to authenticate users to that directory. This involves using the appropriate plug-ins. For example, if you are using Microsoft Active Directory as both the central enterprise directory and the central password store, then you must enable applications to authenticate users to Microsoft Active Directory. You do this by using an external authentication plug-in.

Note: Oracle components use password verifiers to authenticate users, and, when passwords are stored in the third-party directory, those verifiers are not stored in Oracle Internet Directory. On the other hand, if a password is modified by using an Oracle component, then the verifiers are both generated and stored in Oracle Internet Directory.

Advantages and Disadvantages of Storing the Password in Both Directories

If you decide to store the password in both directories, then passwords need to be synchronized, ideally in real-time.

In Oracle Internet Directory 10*g* Release 2 (10.1.2), passwords are not synchronized in real time, but according to a schedule. This can mean an observable delay between the time the password is changed in the central directory and the time that the change is recorded in the other directory.

In deployments with Oracle Internet Directory as the central directory, password values are synchronized regularly from Oracle Internet Directory to the connected directory. This requires you to enable both the password policy of the realm and reversible encryption.

See Also:

- The chapter in *Oracle Internet Directory Administrator's Guide* about password policies for information about setting password policies
- The chapter in *Oracle Internet Directory Administrator's Guide* about directory storage of password verifiers for information about reversible encryption

In general, password values are hashed. If both directories use the same hashing algorithm, then the hashed values can be synchronized as they are. For example, suppose that you have an environment in which SunONE Directory Server and Oracle Internet Directory are integrated. Both of these directories support common hashing algorithms. Now, if the passwords are hashed and stored in SunONE Directory Server by using a hashing technique supported by Oracle Internet Directory, then synchronizing them from SunONE Directory Server to Oracle Internet Directory is the same as with any other attribute.

However if both directories do not support same hashing algorithm, then passwords must be synchronized in clear text format only. For security reasons, password synchronization is possible with Oracle Internet Directory only in SSL Mode 2—that is, server-only authentication.

If Oracle Internet Directory is the source of truth, and if the hashing algorithm it supports is not supported by the other directory, then synchronization is still possible through SSL mode 2 (`sslmode=2`) when reversible password encryption is enabled.

If Microsoft Active Directory is the source of truth, then, when a password is modified in Microsoft Active Directory, a plug-in intercepts the password changes and stores the modified password in a new attribute, preferably in an encrypted form. That attribute can then be synchronized to Oracle Internet Directory. A similar process is required if Oracle Internet Directory is the central enterprise directory and central password store.

Note: In deployments where both directories do not use the same hashing algorithm, password synchronization is not available in an out-of-the-box installation of Oracle Internet Directory. You must configure it.

In deployments where Oracle Internet Directory is not the central directory, the password policy is enforced by the third-party directory. When there is an authentication request to the third-party directory, the latter replies that the authentication either succeeded or failed. However, any detailed password policy errors from the third-party directory are not delivered to Oracle Internet Directory and then to the client applications.

See Also: The following chapters for more detailed information about password synchronization:

- Chapter 20, "Integration with SunONE (iPlanet) Directory Server"
- Chapter 18, "Integration with the Microsoft Active Directory Environment"

The following chapter for information about plug-ins:

- The chapter in *Oracle Internet Directory Administrator's Guide* about the directory plug-in framework
- The chapter in *Oracle Internet Directory Administrator's Guide* about customizing the external authentication plug-in

Choose the Structure of the Directory Information Tree

At installation, each directory server creates a default domain and a default **directory information tree (DIT)** structure. The Oracle Internet Directory infrastructure

installation creates a default realm with designated containers for storing enterprise users and groups. When integrating with a third-party directory, you must create an identical DIT structures in both directories in order to use the default installation of Oracle Internet Directory. Alternatively, you can perform domain-level mapping.

This section contains these topics:

- Create Identical DIT Structures on Both Directories
- Distinguished Name Mapping and Limitations

Create Identical DIT Structures on Both Directories

Oracle Corporation recommends that you configure identical DITs on both directories. This enables all the user and group objects to be synchronized as they are, and spares you the cumbersome task of mapping entries with distinguished names in one directory to URLs in the other. It also spares you the performance problems that such mapping can cause.

To create identical DITs, first decide which directory is the central enterprise directory, and then change the DIT of the other one to match. Be sure to update the directory integration and provisioning profile to reflect the domain level rules.

To enable users to access Oracle applications through Oracle Application Server Single Sign-On, Oracle Corporation recommends that you identify the DIT as a separate identity management realm with its own authentication and authorization domain.

> **See Also:** The chapter on deploying identity management realms in *Oracle Internet Directory Administrator's Guide*

Distinguished Name Mapping and Limitations

If it is not feasible to have identical DITs on both directories, then you need to map the domains between Oracle Internet Directory and the connected directory. For example, suppose that all entries under the container `dc=mydir,dc=com` must be synchronized under `dc=myoid,dc=com` in Oracle Internet Directory. To achieve this, you specify it in the domain level mapping rules.

If the objective is to synchronize all users and groups, then all user entries can be synchronized with appropriate DN mapping. However, group entry synchronization may be both time consuming and carry some additional limitations. This section provides examples of both user and group synchronization when there is a DN mapping.

Example: User Entry Mapping

Suppose that, in a mapping file, the entries in the SunONE Directory Server have the format `uid=name,ou=people,o=iplanet.org`. Suppose further that the entries in Oracle Internet Directory have the format `cn=name,cn=users,dc=iplanet,dc=com`. Note that the naming attribute on SunONE Directory Server is `uid`, but on Oracle Internet Directory it is `cn`.

The mapping file has rules like these:

```
DomainRules
ou=people,o=iplanet.org: cn=users,dc=iplanet,dc=com: cn=%, cn=users,
dc=iplanet,dc=com
AttributeRules
Uid:1: :person:cn: :inetorgperson:
```

The value of 1 in the second column of the last line indicates that, for every change to be propagated from SunONE Directory Server to Oracle Internet Directory, the uid attribute must be present. This is because uid must always be available for constructing the DN of the entry in Oracle Internet Directory.

Example: Group Entry Mapping

When there is a DN mapping, synchronizing group entries is somewhat complex. The group memberships, which are DNs, must have valid DN values after synchronization. This means that whatever DN mapping was done for user DNs must be applied to group membership values.

For instance, suppose that the user DN values are mapped as follows:

```
ou=people,o=iplanet.org: cn=users,dc=iplanet,dc=com:
```

This implies that all the user entries under ou=people,o=iplanet.org are moved to cn=users,dc=iplanet,dc=com.

Group memberships need to be mapped as follows:

```
uniquemember: : : groupofuniquenames: uniquemember: :groupofuniquenames:
dnconvert(uniquemember)
```

For example, if the value of uniquemember is cn=testuser1,ou=people,o=iplanet.org, then it becomes cn=testuser1,cn=users,dc=iplanet,dc=com.

Moreover, if the value of uniquemember is cn=testuser1,dc=subdomain,ou=people,o=iplanet.org, then it becomes cn=testuser1,dc=subdomain,cn=users,dc=iplanet,dc=com.

This is a feasible solution as long as the naming attribute or RDN attribute remains the same on both the directories. However, if the naming attribute is different on different directories—as, for example, ou=people,o=iplanet.org:cn=users,dc=iplanet,dc=com:cn=%,cn=users,dc=iplanet,dc=com—then deriving the actual DNs for group memberships is not achievable through the given set of mapping rules. In this case, DN mapping for the uniquemember or other DN type attributes is not currently feasible.

If you want to synchronize group memberships, remember to keep the naming attribute in the source and destination directories the same.

> **See Also:** "Configuring Mapping Rules" on page 6-3 for instructions on how to specify a mapping rule

Select the Attribute for the Login Name

The attribute for the login name contains the identity of the end user when logging into any Oracle component. It is stored in Oracle Internet Directory as the value of the attribute orclcommonnicknameattribute, under the container cn=common,cn=products,cn=oracleContext,*identity_management_realm*.

By default, orclcommonnicknameattribute has uid as its value. This means that the identity used for login is stored in the uid attribute of the user entry.

If the connected directory has a specific attribute for login, then that attribute needs to be mapped to the right orclcommonnicknameattribute in Oracle Internet Directory. This needs to be one of the mapping rules in the mapping file for the connector associated with synchronizing with the third-party directory.

For example, suppose that you are synchronizing Oracle Internet Directory with Microsoft Active Directory, and that, in the latter, the login identifier is contained in the `userPrincipalName` attribute of the user entry. You would synchronize the value of the `userPrincipalName` attribute to Oracle Internet Directory, storing it in the `uid` attribute, which is the value of the `orclcommonnicknameattribute` attribute. This mapping needs to be reflected in the mapping rules in the directory integration profile.

You can also use any other attribute for login. For example, if you want to use `employeeID` for logins, then mapping rules can be set accordingly. Doing this does not affect your configuration.

Note: The `orclcommonnicknameattribute` attribute is used extensively by Oracle Application Server Single Sign-On, so be sure to plan carefully how you intend to map the attribute to a third-party directory attribute. After you modify this attribute, you must refresh Oracle Application Server Single Sign-On in order for the change to take effect.

See Also: The *Oracle Identity Management Guide to Delegated Administration* for instructions on setting the attribute for login name

Select the User Search Base

The user search context is represented by a multivalued attribute that lists all the containers under which users exist. Depending on your deployment, either set the user search context value to cover the entire user population, or add the container to the user search context attribute by using the Oracle Internet Directory Self-Service Console.

See Also: The *Oracle Identity Management Guide to Delegated Administration* for instructions on setting the user search context

Select the Group Search Base

The group search context is represented by a multivalued attribute that lists all the containers under which groups exist. Depending on your deployment, either set the group search context value to cover all group entries, or add the container to the group search context attribute by using the Oracle Internet Directory Self-Service Console.

See Also: The *Oracle Identity Management Guide to Delegated Administration* for instructions on setting the group search context

Decide How to Address Security Concerns

There are three main security concerns you need to consider:

- Access policies—The user and group search bases should be appropriately protected from the access of any malicious users.
- Synchronization—You can configure the Oracle directory integration and provisioning server to use SSL when connecting to Oracle Internet Directory and third-party directories. If you do this, then all information exchanged between the directory servers is secure.

- Password synchronization—Depending on the configuration, passwords can be synchronized. For instance, when Oracle Internet Directory is the central enterprise directory, password changes can be communicated to the connected directory.

  If passwords are to be synchronized, then Oracle Corporation recommends that you configure communication between the directories in SSL with server-only authentication. The sequence of steps to configure communication between connected directories in SSL is as follows:

  1. In the integration profile, to indicate that the mode of communication is SSL, configure the `connectedDirectoryURL` attribute in the form of `host:port:1`. Make sure the port number is the SSL port. The default SSL port number is 636.
  2. Generate a certificate from the connected directory. What is required is the trust point certificate from the server. You do not need to use any external certificate server to do this.
  3. Export the certificates to Base 64 encoded format.
  4. Import the certificates as trust points in the Oracle Wallet by using Oracle Wallet Manager.
  5. Specify the wallet location in the `odi.properties` file in $*ORACLE_HOME*/ldap/odi/conf.
  6. Store the wallet password by using the Directory Integration and Provisioning Assistant with the `wp` option.
  7. Start the Oracle directory integration and provisioning server in SSL mode.

Step-by-Step Guide to Configuring Synchronization with a Third-Party Directory

This section lists the steps in configuring a sample deployment scenario.

Note: "Step 4: Decide Whether to Create a New Identity Management Realm" through "Step 6: Select the Login Identifiers" involve configuring a new identity management realm and setting its parameters. This can affect the behavior of Oracle Application Server Single Sign-On and any other middle-tier application already installed in the environment. Consequently, make careful decisions at each step and verify the behavior of the applications.

See Also: The chapter on deploying identity management realms in *Oracle Internet Directory Administrator's Guide* for more details on identity management realms and their role in Oracle Application Server.

This section contains these topics:

Step 1: Identify the Default Identity Management Realm in Oracle Internet Directory

To identify the default identity management realm in Oracle Internet Directory:

```
ldapsearch -p port -h host -D distinguished_name -w password
-b "cn=common, cn=products,cn=oraclecontext" -s base "objectclass=*"
orcldefaultsubscriber
```

In this sample deployment, the default identity management realm in Oracle Internet Directory is `dc=us,dc=mycompany,dc=com`.

Step 2: Identify the User and Group Search Bases in Oracle Internet Directory

To identify the user and group search contexts in Oracle Internet Directory:

```
ldapsearch -p port -h host -D distinguished_name -w passwd
-b "cn=common, cn=products,cn=oraclecontext, Identity Management Realm"
-s base "objectclass=*"
```

Note down the values for the `orclcommonusersearchbase` and `orclcommongroupsearchbase` attributes. These are the values which are shown in the Oracle Internet Directory Self-Service Console as User Search Context and Group Search Context.

In this sample deployment, the user and group search contexts in Oracle Internet Directory are:

```
orclcommonusersearchbase is : cn=users,  dc=us,dc=mycompany,dc=com
orclcommongroupsearchbase is : cn=groups,  dc=us,dc=mycompany,dc=com
```

Step 3: Identify the Naming Context on the Remote Directory

The default naming context is the root of the naming context under which the users are stored. Each directory has its own way of creating a default naming context.

If you are using Microsoft Active Directory, then you identify the default naming context by performing the following ldapsearch against that directory:

```
ldapsearch -p port -h host -D distinguished_name -w password -b "" -s base
"objectclass=*" defaultnamingcontext
```

Typically the DNs of users in Microsoft Active Directory are of the form `cn=user name, cn=users, defaultnamingcontext`.

Note that the users also can bind with names such as, `username@domain`.

For example, if the domain name is `newcompany.com`, then the default naming context is `dc=newcompany,dc=com`. The typical login identifier of a user is `user@newcompany.com`.

If you are using SunONE Directory Server, then you identify the naming contexts in that directory by performing the following ldapsearch against it:

```
ldapsearch -p port -h host -D distinguished_name -w password -b "" -s base
"objectclass=*" namingcontexts
```

Different sets of user entries reside in different subtrees. Choose the naming context that contains the objects to be synchronized.

Step 4: Decide Whether to Create a New Identity Management Realm

If the DITs on Oracle Internet Directory and the third-party directory are different, then it is better to create a new identify management realm and make it the default realm. Do this by using either the Oracle Internet Directory Self-Service Console or the Oracle Internet Directory Configuration Assistant. On the other hand, if the third-party directory is Microsoft Active Directory in which the default naming context is `mycompany.com`, then you may not have to create the new identity management realm.

Step 5: Select the User Search Base and Group Search Base

How you do this depends on whether you created a new identity management realm as discussed in the previous step.

If a new identity management realm has been created, then:

1. Select the user search base and the user creation context. Do this by using the Oracle Internet Directory Self-Service Console. Set the user search context to reflect the container under which users are stored in the third-party directory. This is described in the *Oracle Identity Management Guide to Delegated Administration*.

   Follow the same approach to set the user creation context.

2. Select the group search base and the group creation context. Do this by using the Oracle Internet Directory Self-Service Console. Set the group search context to reflect the container under which groups are stored in the third-party directory. This is described in the *Oracle Identity Management Guide to Delegated Administration*.

   Follow the same approach to set the group creation context.

If a new identity management realm has not been created, then, to enable user and group entries to be accessed by all Oracle components, you must modify the default parameters in the Oracle Internet Directory Self-Service Console. To do this:

1. In the User Search Context, enter the DN of the users container in the third-party directory, or enter the subtree of the containers specified in the search context. For example, enter either of the following:

   `cn=users,dc=myCompany,dc=com`

`dc=myCompany,dc=com`.

2. In the Group Search Context, either enter the DN of the groups container in the third-party directory, or enter the subtree of the containers specified in the search context. For example, enter either of the following:

   `cn=groups,dc=myCompany,dc=com`

   `dc=myCompany,dc=com`

   See Also: The *Oracle Identity Management Guide to Delegated Administration*

Step 6: Select the Login Identifiers

The attribute used for login is `orclcommonnicknameattribute`. In the Oracle Internet Directory Self-Service Console, the field is named Attribute for Login Name. The default value is `UID`. Oracle Corporation recommends that you keep the default value. If this attribute is modified—for example, if it is changed to `mail`—then be sure that all entries under the container that you are working with have the `mail` attribute value populated. Otherwise, the user cannot login through Oracle Application Server Single Sign-On.

Step 7: Modify the Mapping File to Reflect the Changes You Have Made

The attributes you have just modified can require a change in the default mapping files. Look carefully at the various mapping rules and modify them according to the requirements. If the users and groups are under different containers, you may need to specify multiple set of domain rules in the same mapping file.

Default mapping rules for integration with SunONE Directory Server and Microsoft Active Directory are in the directory $*ORACLE_HOME*/ldap/odi/conf.

The important parameters to be modified are:

- Mapping rule for the `loginid` attribute
  - In the default profile for Microsoft Active Directory, the default mapping rule for the `loginid` attribute in the sample mapping file is:

    ```
    Userprincipalname: : :user: uid: : :inetorgperson
    ```

  - In the default profile for SunONE Directory Server, the `UID` is directly mapped to the `UID` attribute.

    This can be modified depending on which attribute is used for login. For example, to use `employeenumber` as the `loginid`, modify the mapping rule as follows:

    ```
    Employeenumber: : :user: uid: : :inetorgperson
    ```

- Mapping rule for the Kerberos login—To support Windows native authentication, Oracle Application Server Single Sign-On uses Kerberos login for the Windows environment. In such cases, a mapping rule is required for the Windows login. The attribute for the Kerberos login is `orclcommonkrbprincipalattribute` in the entry `cn=common,cn=public,cn=oraclecontext,`*`identity_management_realm`*. By default, it is set to `krbPrincipalName`.

  For integration with Microsoft Active Directory, the default mapping rule is:

  ```
  Userprincipalname: : :user: krbPrincipalName: : :orclUserV2.
  ```

This rule maps the user principal name in Microsoft Active Directory to the Kerberos principal name. To support another value for Kerberos login, modify this rule.

> **See Also:** *Oracle Application Server Single Sign-On Administrator's Guide* for information about support for Windows native authentication in Oracle Application Server Single Sign-On

Step 8: Create or Modify the Synchronization Profile with the New Set of Mapping Rules

To do this, use the Directory Integration and Provisioning Assistant.

```
dipassistant mp -profile profile_name odip.profile.mapfile=relative_path_name_of_
mapping_file
```

Step 9: Configure Access Control

Configure access control to various containers in either of the following:

- The profile `orclodipagentname=profile_name,cn=subscriber profile,cn=changelog subscriber,cn=oracle internet directory`'
- The group `cn=odipgroup,cn=odi,cn=oracle internet directory`

A sample ACI is available in $*ORACLE_HOME*/ldap/odi/samples/commonaci.ldif. This sample contains the following attributes, all of which have the same values:

- `UserSearchBase`
- `GroupSearchBase`
- `UserCreateBase`
- `GroupCreateBase`

You can use Oracle Directory Manager to set ACIs to these containers.

Step 10: Bootstrap the Directory by Using the Directory Integration and Provisioning Assistant

To bootstrap the directory, use the `bootstrap` command in the Directory Integration and Provisioning Assistant.

> **See Also:**
>
> - Chapter 8, "Bootstrapping of a Directory in Oracle Directory Integration and Provisioning"
> - The `dipassistant` section of the Oracle Directory Integration and Provisioning tools chapter of the *Oracle Identity Management User Reference* for instructions on using the `bootstrap` command of the Directory Integration and Provisioning Assistant

Step 11: Update the Last Change Number for Synchronization

To do this, enter:

```
dipassistant mp -profile profile_name -updlcn
```

The Directory Integration and Provisioning Assistant determines the connected directory by reading the directory integration profile.

Step 12: Enable the Profile by Using Either the Oracle Directory Integration and Provisioning Server Administration Tool or the Directory Integration and Provisioning Assistant

You can do this by using either the Oracle Directory Integration and Provisioning Server Administration tool or the Directory Integration and Provisioning Assistant.

See Also:

- "Creating a Profile by Using the Oracle Directory Integration and Provisioning Server Administration Tool" on page 7-1 for instructions on doing this by using the Oracle Directory Integration and Provisioning Server Administration tool
- "Managing Synchronization Profiles by Using Command-Line Tools" on page 7-3 for instructions on doing this by using the Directory Integration and Provisioning Assistant

Step 13 (Optional): Enable the External Authentication Plug-in for Password Synchronization

If you need to synchronize password changes from Oracle Internet Directory to the third-party directory, then enable the external authentication plug-in by doing the following:

- Enable the password policy in the identity management realm. You can do this by using either the Oracle Internet Directory Self-Service Console or Oracle Directory Manager.
- Enable reversible password encryption by setting the `orclpwdencryptionenable` attribute to `TRUE`.

When passwords are synchronized to directories that do not support the hashing technique used by Oracle Internet Directory, synchronization can be done only by using the SSL mode 2 (`sslmode=2`).

See Also:

- The section in *Oracle Internet Directory Administrator's Guide* about managing password policies by using the Self-Service Console
- The chapter on password policies in *Oracle Internet Directory Administrator's Guide*
- The chapter on directory storage of password verifiers in *Oracle Internet Directory Administrator's Guide* for information about enabling reversible encryption

Step 14: Start the Oracle Directory Integration and Provisioning Server

Do this by following the instructions in "Starting, Stopping, and Restarting the Oracle Directory Integration and Provisioning Server" on page 4-8.

Note: To synchronize passwords, start Directory Integration and Provisioning with `sslmode=2`—that is, server-only authentication.

Limitations of Third-Party Integration in Oracle Internet Directory 10*g* Release 2 (10.1.2)

Oracle Internet Directory 10*g* Release 2 (10.1.2) does not support the synchronization of the schema and ACLs. If you are changing the schema or ACLs, then you must apply the changes manually. Use the `schemasync` tool to synchronize the schema between Oracle Internet Directory and a third-party directory.

> **See Also:** The `schemasync` section in the Oracle Directory Integration and Provisioning tools chapter of the *Oracle Identity Management User Reference*

18

Integration with the Microsoft Active Directory Environment

This chapter explains how Oracle Identity Management can integrate with Microsoft Active Directory in a production environment.

Note: This chapter assumes familiarity with the chapter on Oracle Internet Directory concepts and architecture in the *Oracle Internet Directory Administrator's Guide*. It also assumes familiarity with the earlier chapters in this book, especially:

- Chapter 1, "Introduction to Oracle Identity Management Integration"
- Chapter 4, "Managing the Oracle Directory Integration and Provisioning Server"
- Chapter 5, "Oracle Directory Synchronization Service"
- Chapter 17, "Considerations for Integrating with Third-Party Directories"

If you are configuring a demonstration of integration with Microsoft Active Directory, then see the Oracle By Example series for Oracle Identity Management Release 10*g* Release 2 (10.1.2), available on Oracle Technology Network at
`http://www.oracle.com/technology/`

This chapter contains these topics:

See Also:

- Chapter 19, "Integration with the Microsoft Windows NT 4.0 Environment" for information about integrating with the Microsoft Windows domain database
- "Troubleshooting Integration with Microsoft Active Directory" on page C-25
- "Oracle Internet Directory Frequently Asked Questions" on the Oracle Technology Network at `http://www.oracle.com/technology`
- *Oracle Identity Management Guide to Delegated Administration* for instructions on how to use the Oracle Internet Directory Self-Service Console

Concepts and Architecture of Microsoft Active Directory Integration

Oracle provides centralized security administration for all Oracle components by integrating them with Oracle Identity Management. Similarly, Microsoft provides centralized security administration in Microsoft Windows by integrating all Microsoft applications with Microsoft Active Directory.

If your environment uses both Oracle Identity Management and Microsoft Active Directory, then, to synchronize data in one with data in the other, you need to integrate the two systems. You do this by using Active Directory Connector.

This section discusses the Oracle components and architecture involved in integrating Oracle Identity Management with Active Directory. It contains these topics:

- Components for Integrating with Microsoft Active Directory
- How Oracle Directory Integration and Provisioning Maintains Synchronization
- Oracle Internet Directory Schema Elements for Integration with Microsoft Active Directory
- Directory Information Tree in an Integration with Microsoft Active Directory

Components for Integrating with Microsoft Active Directory

This section describes the following components that are used to integrate with Microsoft Active Directory:

- Oracle Internet Directory
- Oracle Directory Integration and Provisioning
- Oracle Application Server Single Sign-On
- Active Directory External Authentication Plug-in
- Windows Native Authentication

See Also: Chapter 3, "Oracle Directory Integration and Provisioning Administration Tools" for a description of the tools used to integrate Oracle Internet Directory with Microsoft Active Directory

Oracle Internet Directory

Oracle Internet Directory is the repository in which Oracle components and third-party applications store and access user identities and credentials. It uses the

Oracle directory server to authenticate users by comparing the credentials entered by users with the credentials stored in Oracle Internet Directory. When credentials are stored in a third-party directory and not in Oracle Internet Directory, users can still be authenticated. In this case, Oracle Internet Directory uses an external authentication plug-in that authenticates users against the third-party directory server.

See Also:

- The chapter on security in *Oracle Internet Directory Administrator's Guide* for a discussion of security in Oracle Internet Directory
- "Active Directory External Authentication Plug-in" on page 18-4 for a brief discussion of the external authentication plug-in

Oracle Directory Integration and Provisioning

Oracle Directory Integration and Provisioning is installed as part of the Oracle Application Server infrastructure. You can configure it to run on the same host as Oracle Internet Directory or on a different host.

Oracle Directory Integration and Provisioning enables:

- Synchronization between Oracle Internet Directory and other directories and user repositories
- Automatic provisioning services for Oracle components

Oracle Directory Integration and Provisioning includes connectors to synchronize Oracle Internet Directory with other LDAP directories or data stores. One of its connectors, Active Directory Connector, is designed to synchronize Oracle Internet Directory with Microsoft Active Directory.

Active Directory Connector enables you to:

- Configure either one-way or two-way synchronization with Microsoft Active Directory
- Designate a specific subset of attributes for synchronization. You do this by configuring the appropriate mapping rules, which you can then change at run time
- Synchronize with multiple Microsoft Active Directory domains. You can synchronize changes with an individual domain or an entire Active Directory environment by using the Microsoft Global Catalog.

See Also:

- Chapter 19, "Integration with the Microsoft Windows NT 4.0 Environment" for instructions on synchronizing with a Microsoft Windows NT domain database
- "Attribute-Level Mapping" on page 6-5 for a discussion about configuring attribute mapping rules

Oracle Application Server Single Sign-On

OracleAS Single Sign-On enables users to access Oracle Web-based components by logging in only once.

Oracle components delegate the login function to the OracleAS Single Sign-On server. When a user first logs in to an Oracle component, the component directs the login to the OracleAS Single Sign-On server. The OracleAS Single Sign-On server compares the credentials entered by the user to those stored in Oracle Internet Directory. After

verifying the credentials, the OracleAS Single Sign-On server grants the user access to all components the user is authorized to use throughout the current session.

Oracle Application Server Single Sign-On enables native authentication in a Microsoft Windows environment. Once logged in to the Windows environment, the user automatically has access to Oracle components. OracleAS Single Sign-On automatically logs in the user to the Oracle environment using the user's Kerberos credentials.

See Also:

- *Oracle Application Server Single Sign-On Administrator's Guide* for information about OracleAS Single Sign-On
- The sections "Windows Native Authentication" on page 18-4 and "Configuring Windows Native Authentication" on page 18-39 for discussions on using Windows native authentication and how to configure it when integrating with Microsoft Active Directory

Active Directory External Authentication Plug-in

This plug-in, which is part of the Oracle directory server, enables Microsoft Windows users to log in to the Oracle environment by using their Microsoft Windows credentials. When this plug-in is in place, it is invoked by the Oracle directory server. This plug-in verifies the user's credentials in Microsoft Active Directory. If the verification is successful, then the Oracle directory server notifies OracleAS Single Sign-On.

See Also: "Configuring the Active Directory External Authentication Plug-in" on page 18-37

Windows Native Authentication

Windows native authentication is an authentication scheme for users of Microsoft Internet Explorer on Microsoft Windows. When this feature is enabled in OracleAS Single Sign-On, users log in to OracleAS Single Sign-On partner applications automatically. To do this, they use Kerberos credentials obtained when the user logged in to a Windows domain.

Using the Simple and Protected GSS-API Negotiation Mechanism (SPNEGO) protocol, Internet Explorer version 5.0 and later can automatically pass the user's Kerberos credentials to a requesting Kerberos-enabled Web server. The Web server can then decode the credentials and authenticate the user.

Although the SPNEGO protocol supports both Kerberos version 5 and NT Lan Manager (NTLM) authentication schemes, Oracle Application Server 10*g* Release 2 (10.1.2) supports only Kerberos V5 with SPNEGO.

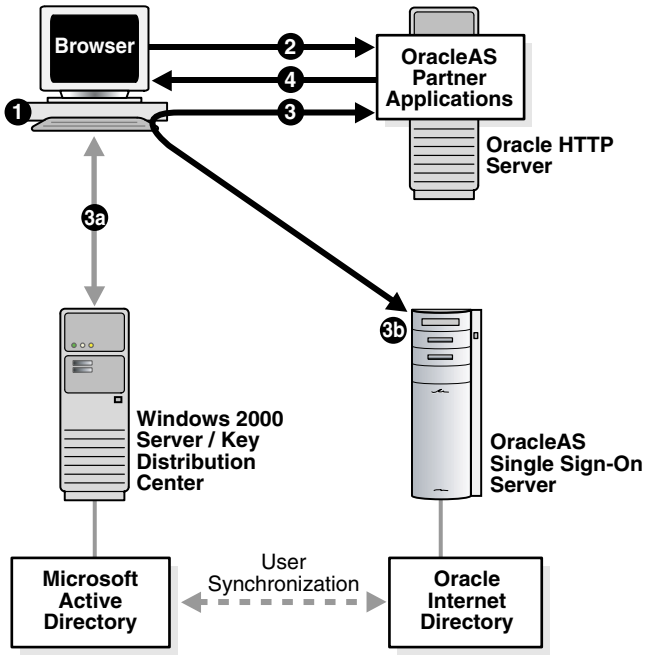
Note: Although this chapter refers only to Windows 2000, Windows native authentication is also supported on the Windows XP platform.

If the browser is not Internet Explorer 5.0, then Oracle Identity Management authenticates the user by using OracleAS Single Sign-On. Authentication to Active Directory is performed by using the Active Directory external authentication plug-in.

The following steps, shown in Figure 18–1 on page 18-5, describe what happens when a user tries to access a single-sign-on-protected application:

1. The user logs in to a Kerberos realm, or domain, on a Windows computer.
2. The user attempts to access a single-sign-on partner application using Internet Explorer.
3. The application routes the user to the single sign-on server for authentication. As part of this routing, the following occurs:
   a. The browser obtains a Kerberos session ticket from the Key Distribution Center (KDC).
   b. The OracleAS Single Sign-On server verifies the Kerberos session ticket and, if the user is authorized, then the user is allowed to access the requested URL.
4. The application provides content to the user.

Figure 18–1 Flow for Windows Native Authentication



When the user logs out of the Windows session, this application and any single sign-on applications accessed are logged out at the same time.

> **See Also:** "Configuring Windows Native Authentication" on page 18-39

How Oracle Directory Integration and Provisioning Maintains Synchronization

To keep Oracle Internet Directory and Microsoft Active Directory synchronized, Oracle Directory Integration and Provisioning brings in incremental changes made available by Microsoft Active Directory change tracking mechanisms. Oracle Directory Integration and Provisioning supports two of these mechanisms:

- The DirSync approach, which uses an LDAP control that is supported by Microsoft Active Directory
- The USN-Changed approach, which uses an attribute of the entry

In each approach, the directory from which changes are derived is queried at scheduled intervals by Active Directory Connector.

Each approach has advantages and disadvantages. Table 18–1 compares the two approaches.

Table 18–1 Comparing the DirSync Approach to the USN-Changed Approach

Considerations	DirSync Approach	USN-Changed Approach
Change key	Presents changes to the `ObjectGUID`, the unique identifier of the entry	Presents changes to the distinguished name. The `ObjectGUID` is used to keep track of modifications of the DN.
Error handling	If synchronization stops as a result of an error condition, then, during the next cycle, all changes that are already applied are read and skipped.	Does not require synchronization to be atomic. If synchronization stops, then the next synchronization cycle starts from the entry where the synchronization was interrupted.
Information in the search results	Changes consist of only the changed attributes and the new values. This can be quicker than the USN-Changed approach.	All attributes of the changed entry are retrieved. The retrieved values are compared to the old values stored in Oracle Internet Directory and updated. This can be more time consuming than the DirSync approach.
Changes to multivalued attributes	Reflects incremental changes made to multivalued attributes as a complete replacement of the attribute value.	Reflects incremental changes made to multivalued attributes as a complete replacement of the attribute value.
How synchronization point is tracked	When queried for changes in the directory, presents incremental changes based on a cookie value that identifies the state of the directory.	The changes are queried in the directory based on the `uSNChanged` attribute, which is a long integer, that is, 8 bytes. You can modify the value to adjust where to start the synchronization.
Required user privileges	Requires the user to have the "Replicate Changes" privilege on the naming context of interest. This enables reading all objects and attributes in Microsoft Active Directory regardless of the access protections on them. See Also: ■ "Configuring the Connection Details for Microsoft Active Directory" on page 18-19 ■ The Microsoft Knowledge Base Article 303972 available at `http://support.microsoft.com/` for instructions on how to assign privileges to Microsoft Active Directory users when using the DirSync approach. Apply to this context the instructions used for Active Directory management agent in this article.	Requires the Microsoft Active Directory user to have the privilege to read all required attributes to be synchronized to Oracle Internet Directory. See Also: Microsoft networking and directory documentation available in the Microsoft library at the following URL: `http://msdn.microsoft.com/` for instructions about how to assign privileges to Microsoft Active Directory users when using the USN-Changed approach.
Support of multiple domains	Requires separate connections to different domain controllers to read changes made to the entries in different domains.	Can obtain changes made to the multiple domains by connecting to the Global Catalog server. See Also: "Considerations for Synchronizing with a Multiple-Domain Microsoft Active Directory Environment" on page 18-27

Table 18–1 (Cont.) Comparing the DirSync Approach to the USN-Changed Approach

Considerations	DirSync Approach	USN-Changed Approach
Synchronization from a replicated directory when switching to a different Microsoft Active Directory domain controller	Synchronization can continue. The synchronization key is the same when connecting to a replicated environment.	Requires: ■ Full synchronizing to a known point ■ Updating the `uSNChanged` value ■ Starting synchronization with the failover directory See Also: "Switching to a Different Microsoft Active Directory Domain Controller in the Same Domain" on page 18-54
Synchronization scope	Reads all changes in the directory, filters out changes to the required entries, and propagates to Oracle Internet Directory	Enables synchronization of changes in any specific subtree
Usability in an environment with multiple Microsoft Active Directory servers behind a load balancer	-	Either connect to a specific Microsoft Active Directory domain controller, or connect to a Global Catalog. Connect to Global Catalog if: ■ You are interested in import operations only. ■ The Global Catalog contains all entries and attributes to be synchronized. ■ Performance of the Global Catalog is acceptable.

Oracle Internet Directory Schema Elements for Integration with Microsoft Active Directory

To identify objects that are synchronized with those in Microsoft Active Directory, Oracle Internet Directory contains schema elements that correspond to Active Directory-specific attributes. These schema elements are described in the *Oracle Identity Management User Reference*.

Directory Information Tree in an Integration with Microsoft Active Directory

This section contains the following topics:

- About Realms in Oracle Internet Directory
- Planning the Deployment
- Example: Integration with a Single Microsoft Active Directory Domain Controller
- Example: Integration with Multiple Microsoft Active Directory Domain Controllers

> **See Also:** The chapter on directory concepts and architecture in *Oracle Internet Directory Administrator's Guide* for a fuller discussion of directory information trees.

About Realms in Oracle Internet Directory

In Oracle Internet Directory, an identity management realm defines an enterprise scope over which certain identity management policies are defined and enforced by the deployment. It comprises:

- A well-scoped collection of enterprise identities—for example, all employees in the US domain.

- A collection of identity management policies associated with these identities. An example of an identity management policy would be to require that all user passwords have at least one alphanumeric character.
- A collection of groups, that is, aggregations of identities that simplify setting the identity management policies

Multiple Realms

You can define multiple identity management realms within the same Oracle Identity Management infrastructure. This enables you to isolate user populations and enforce a different identity management policy,—for example, password policy, naming policy, self-modification policy—in each realm. This is useful in a hosted deployment of Oracle Application Server.

Each identity management realm is uniquely named to distinguish it from other realms. It also has a realm-specific administrator with complete administrative control over the realm.

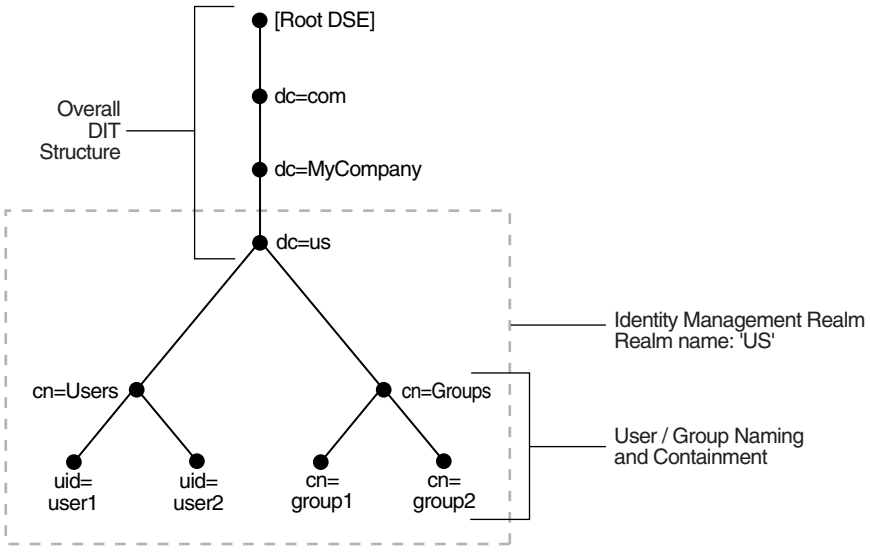
The Default Realm

For all Oracle components to function, an identity management realm is required. One particular realm, created during installation of Oracle Internet Directory, is called the default identity management realm. It is where Oracle components expect to find users, groups, and associated policies whenever the name of a realm is not specified. This default realm facilitates proper organization of information and enforces proper access controls in the directory.

There can be only one default identity management realm in the directory. If a deployment requires multiple identity management realms, then one of them must be chosen as the default.

Figure 18–2 illustrates the default identity management realm.

Figure 18–2 The Default Identity Management Realm



As Figure 18–2 shows, the default identity management realm is part of a global DIT. The node that follows the root DSE is `dc=com`, followed by `dc=MyCompany`, then `dc=us`. These four nodes represent the overall DIT structure. The node `dc=us` is the root of the default identity management realm. It has two subtrees for containing user

and group information: `cn=Users` and `cn=Groups`. For illustration purposes, the `cn=Users` node contains two leaves: `uid=user1` and `uid=user2`. Similarly, the `cn=Groups` node contains `cn=group1` and `cn=group2`.

Access Control Policies in the Realm

You must configure appropriate ACLs in Oracle Internet Directory to enable Oracle Directory Integration and Provisioning to:

- Enable the import profile to add, modify and delete objects in the `users` and `groups` containers. By default, import profiles are part of the Realm Administrators group, which can perform all operations on any entry under the realm DN. If you have customized ACLs in the realm, then be sure that the import profiles have the appropriate privileges to perform these operations on the subtree to be synchronized or on either the `user` container, the group container, or both depending on where the synchronization takes place.
- Enable Oracle components to manage the users and groups in the realm. By default, Oracle components can manage users and groups in the `users` and `groups` containers respectively. If you have updated your `usersearchbase` and `groupsearchbase` in the realm, then set up appropriate ACLs on the `users` container and `groups` container.

  See Also: The chapter on deployment of Oracle Identity Management realms in *Oracle Internet Directory Administrator's Guide* for a description of the default realm installed with Oracle Internet Directory

Planning the Deployment

When planning the DIT, the most important decisions to make before synchronization are:

- Which directory is to be the central one
- What objects to synchronize, for example:
  - The portion of the DIT that you want to synchronize. You can synchronize the entire DIT or just a portion of it.
  - For each entry, the specific contents that you want to synchronize. You can synchronize the entire content of the entry or just a portion of it.
- Where to synchronize. You have two options:
  - You can synchronize so that the relative position of each entry in the DIT is the same in the source and destination directories. This configuration, called one-to-one distinguished name mapping, is the most commonly used configuration. Because the source DN is the same as the destination DN, this configuration provides better performance than when the two DNs are different.
  - You can synchronize so that the relative position in the DIT of each entry in the destination directory is different from that in the source directory. In this configuration, the Oracle directory integration and provisioning server must change the DN values of all entries being mapped, including their references in group entries. This requires more intensive computation.

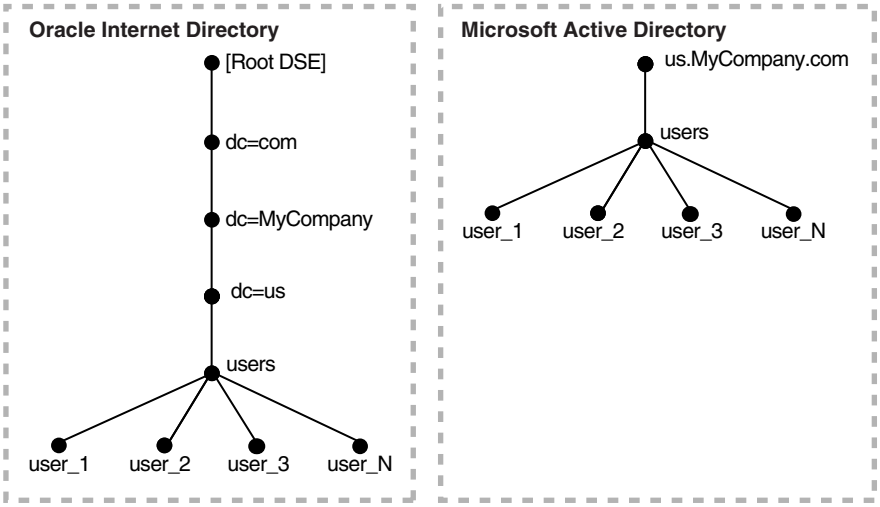
    If you synchronize in this way, you need to use the `dnconvert` mapping rule as described in "Supported Attribute Mapping Rules and Examples" on page 6-8.

See Also: The section "Choose the Structure of the Directory Information Tree" on page 17-6 for more information about planning the directory information tree

Example: Integration with a Single Microsoft Active Directory Domain Controller

Figure 18–3 shows an example of one-to-one mapping between the two directories.

Figure 18–3 Default DIT Structures in Oracle Internet Directory and Active Directory When Both Directory Hosts Are Under the Domain us.MyCompany.com



In the one-to-one mapping illustrated in Figure 18–3:

- Both Active Directory and Oracle Internet Directory hosts have the same topology.
- Users are synchronized only from Active Directory to Oracle Internet Directory. All users to be synchronized are stored in one container in Active Directory, in this case `users.us.MyCompany.com`.
- The same DIT structure is maintained in both Active Directory and Oracle Internet Directory. All users appear in the same users subtree identified by the value `cn=users,dc=us,dc=MyCompany,dc=com`.

In the example shown in Figure 18–3, only the `users` subtree must be synchronized from Active Directory to Oracle Internet Directory using one-to-one domain mappings.

Note: In the example in Figure 18–3, the two directories have the same topology, but be aware that this is for illustration purposes only. The two directories do not need to be in the same domain. Oracle Internet Directory can be anywhere in the network, provided it can connect to Microsoft Active Directory.

In addition, although the synchronization in the example is one-way, from Microsoft Active Directory to Oracle Internet Directory, the synchronization can, alternatively, be bi-directional.

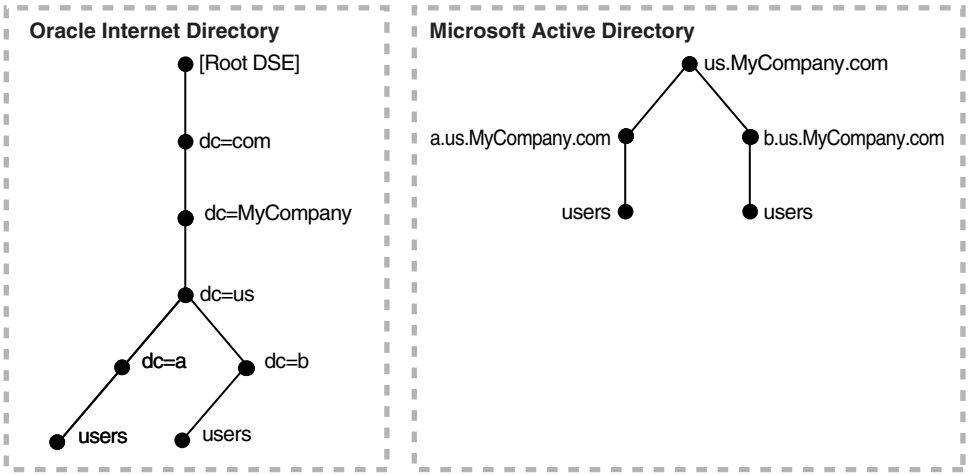
Example: Integration with Multiple Microsoft Active Directory Domain Controllers

A deployment of Microsoft Active Directory with multiple domains can have either a single DIT or a combination of two or more DITs. In Microsoft Active Directory, a group of DITs is called a forest.

One-to-One Mapping of Multiple Microsoft Active Directory Domains

Figure 18–4 shows how multiple domains in Microsoft Active Directory are mapped to a DIT in Oracle Internet Directory.

Figure 18–4 Example of a Mapping Between Oracle Internet Directory and Multiple Domains in Microsoft Active Directory



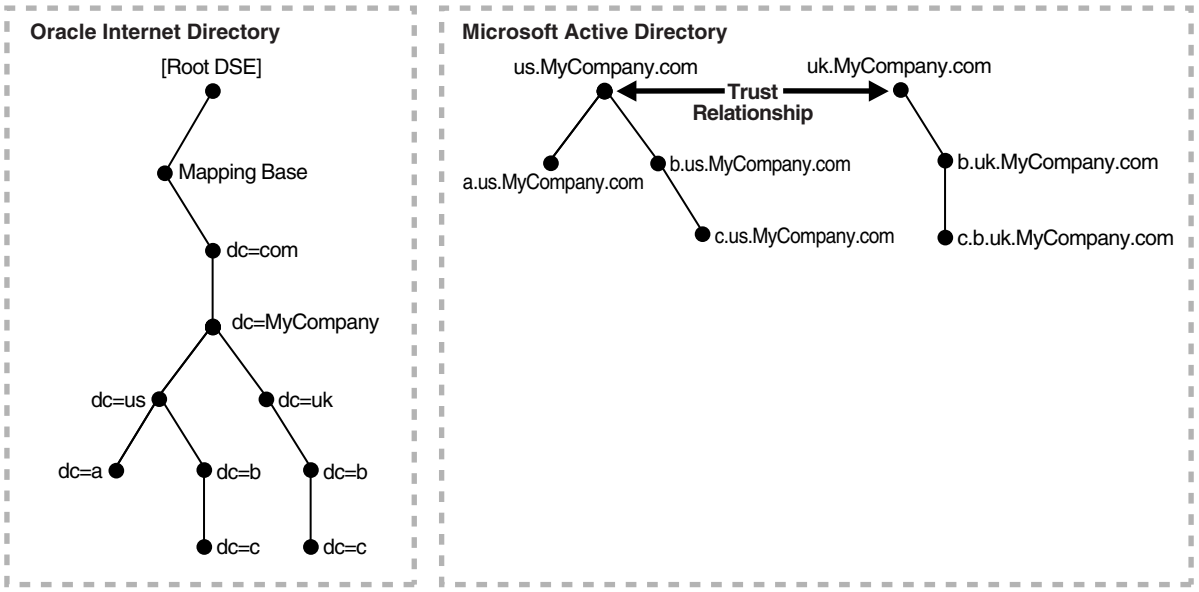
In Figure 18–4, the Microsoft Active Directory environment has a parent and two children. Each domain has a domain controller associated with it. The Active Directory domain controller supporting the node `us.mycompany.com` is the Global Catalog server.

The first child domain `a.us.MyCompany.com` maps to `dc=a,dc=us,dc=MyCompany,dc=com` in Oracle Internet Directory. The second child domain `b.us.MyCompany.com` maps to `dc=b,dc=us,dc=MyCompany,dc=com` in Oracle Internet Directory. The common domain component in Active Directory environment `us.MyCompany.com` maps to the default identity management realm in Oracle Internet Directory, in this case `dc=us,MyCompany,dc=com`.

Mapping of a Microsoft Active Directory Forest

Figure 18–5 shows how a forest in Microsoft Active Directory is reflected in Oracle Internet Directory.

Figure 18–5 Mapping Between Oracle Internet Directory and a Forest in Microsoft Active Directory



In this directory, two domain trees constitute a forest. These trees are in a trust relationship, that is, users in one domain are authenticated by the domain controller in the other domain. This forest in Microsoft Active Directory maps to an identically structured subtree in Oracle Internet Directory.

Foreign Security Principals

A Microsoft Active Directory user or computer account represents a physical entity such as a computer or person. User accounts and computer accounts, as well as groups, are called security principals. Security principals are directory objects that are automatically assigned security identifiers. Objects with security identifiers can log on to the network and access domain resources. A user or computer account is used to:

- Authenticate the identity of the user or computer
- Authorize or deny access to domain resources
- Administer other security principals
- Audit actions performed using the user or computer account

For example, the user and computer accounts that are members of the Enterprise Administrators group are automatically granted permission to log on at all of the domain controllers in the forest.

User and computer accounts are added, disabled, reset, and deleted by using Microsoft Active Directory Users and Computers.

In a trust relationship in Active Directory, users in one domain are authenticated by a domain controller in another domain. The trust relationship can be transitive or nontransitive.

- In a transitive trust relationship, the trust relationship extended to one domain is automatically extended to all other domains that trust that domain. For example, suppose you have three domains: A, B, and C in which both B and C are in a direct trust relationship with A. In this scenario, both B and C also trust each other. This is because, although they are not in a direct trust relationship with each other, they are in a direct trust relationship with A.

- In a nontransitive trust relationship, the trust is bound by the two domains in the trust relationship; it does not flow to any other domains in the forest.

When a trust is established between a Windows 2000 domain in a particular forest and a Windows 2000 domain outside of that forest, security principals from the external domain can be granted access to resources in the forest. A security principal from an external domain is called a *foreign security principal* and is represented in Active Directory as a "foreign security principal" object. These foreign security principals can become members of domain local groups, which can have members from domains outside of the forest.

Foreign security principals are used when there is a nontransitive trust between two domains in a Microsoft Active Directory environment.

In a nontransitive trust relationship in a Microsoft Active Directory environment, when one domain recognizes a foreign security principal from the other domain, it represents that entity similar to a DN entry. In that entry, the RDN component is set to the SID of the original entry in the trusted domain. In the case of groups, the DNs of the foreign security principals are represented as member values, not as the DNs of the original entries in the trusted domain. This can create a problem when foreign security principals are synchronized with Oracle Internet Directory.

Deployment Options for Integrating with Microsoft Active Directory

There are two common ways of integrating with a Microsoft Windows environment:

- Using Oracle Internet Directory as the central directory for user and group data for the Microsoft Windows 2000 and Windows NT environments
- Using Microsoft Active Directory as the central enterprise directory for user and group data for Oracle components

This section discusses the requirements of each deployment. It contains the following topics:

Deployments with Oracle Internet Directory as the Central Directory

Table 18–2 describes the typical requirements in this deployment.

Table 18–2 Typical Requirements with Oracle Internet Directory as the Central Directory

Requirement	Description
Initial startup	The Directory Integration and Provisioning Assistant populates Microsoft Active Directory with users and groups stored in Oracle Internet Directory. If there are multiple Microsoft Active Directory domains, then the Directory Integration and Provisioning Assistant must be run as many times as there are Microsoft Active Directory domains. Each time you do this, you choose the specific data set required by the target Microsoft Active Directory domain.
Synchronization	User and group information is managed in Oracle Internet Directory. Changes to that information are synchronized with Microsoft Active Directory by the Oracle directory integration and provisioning server when an import profile has been configured. Synchronization from Microsoft Active Directory into Oracle Internet Directory can be achieved by configuring an import profile.
Passwords and password verifiers	Passwords are managed in Oracle Internet Directory by using Oracle tools such as the Oracle Internet Directory Self-Service Console. Password changes are synchronized with Microsoft Active Directory by the Oracle directory integration and provisioning server. However, before this server can synchronize the password changes, the password synchronization must be configured in the mapping rules. Because the password is securely managed, the communication for synchronizing passwords to Microsoft Active Directory must be over SSL. Run the Oracle directory integration and provisioning server in the server-only authentication mode with the proper certificate from Microsoft Active Directory. Be sure that Active Directory is also enabled for SSL. If the Oracle environment requires a password verifier, then the password verifier is automatically generated when a new user entry is created or when a password is modified.
Oracle Application Server Single Sign-On	Users log in to the Oracle environment by using the OracleAS Single Sign-On server. When called upon by the OracleAS Single Sign-On server to authenticate a user, the Oracle directory server uses credentials available locally. No external authentication is involved. Users must log in only once to access various components in the Oracle environment.

New users or groups in Oracle Internet Directory can be automatically provisioned into the Microsoft Windows environment by the Oracle directory integration and provisioning server. This automatic provisioning requires that:

- The Oracle directory server is running with the change log enabled
- The change log is not purged

If these two conditions are not met, then you must load the entries in Oracle Internet Directory to an LDIF file and upload the data to Microsoft Active Directory.

If multiple Microsoft Active Directory domains are involved, then the Oracle directory integration and provisioning server provisions users and groups in the respective Microsoft Active Directory domains. Before provisioning can take place, you must configure a one-way synchronization from Oracle Internet Directory to the Microsoft Active Directory domain.

> **See Also:** The chapter on garbage collection in *Oracle Internet Directory Administrator's Guide* for information about purging the change log

Deployments with Microsoft Active Directory as the Central Directory

Table 18–3 describes the typical requirements in this deployment.

Table 18–3 Typical Requirements with Microsoft Active Directory as the Central Directory

Requirement	Description
Initial startup	The Directory Integration and Provisioning Assistant populates Oracle Internet Directory with users and groups stored in Microsoft Active Directory. If there are multiple Microsoft Active Directory servers, then you must bootstrap the data from each Microsoft Active Directory domain. If you use the Global Catalog for one-way synchronization from Microsoft Active Directory to Oracle Internet Directory, then you need to bootstrap only once from the Global Catalog server. You can choose to manage user information, including password credentials, in Microsoft Active Directory only. In such deployments, to enable single sign-on in the Oracle environment, the Oracle directory integration and provisioning server can synchronize only those user entry attributes required by Oracle components. Passwords are not migrated from Microsoft Active Directory to Oracle Internet Directory.
Synchronization	The central directory for user and group information is Microsoft Active Directory. Changes to user and group information in Active Directory are synchronized with Oracle Internet Directory by the Oracle directory integration and provisioning server when an import profile has been configured. Synchronization from Oracle Internet Directory to Microsoft Active Directory is achieved by configuring an export profile.
Passwords and password verifiers	Passwords are managed in typically Active Directory by using Microsoft Windows tools. The Oracle directory integration and provisioning server does not synchronize password changes into Oracle Internet Directory.
Oracle Application Server Single Sign-On	Users log in to the Oracle environment only once by using the OracleAS Single Sign-On server. Users with credentials only in Microsoft Active Directory are authenticated by the Oracle directory server invoking the external authentication plug-in. Users with credentials in Oracle Internet Directory are authenticated locally by the Oracle directory server.
Windows native authentication	Same as in Oracle Internet Directory-centered deployment. However, for a user to use Windows native authentication, a user must exist in Active Directory. If Windows native authentication is enabled, then, for local Oracle Internet Directory users to invoke the single sign-on server, you must populate the attributes `orclsamaccountname` and `krbprincipalname` for each user entry.
Active Directory external authentication plug-in	When user credentials are managed in Microsoft Active Directory, this plug-in is required. To authenticate a user, the OracleAS Single Sign-On server calls upon the Oracle directory server. The plug-in then performs the authentication of the user against the user credentials stored in Active Directory.

New users or groups created in Microsoft Active Directory are automatically synchronized into Oracle Internet Directory by the Oracle directory integration and provisioning server. Before the provisioning can take place, a one-way synchronization between Microsoft Active Directory and Oracle Internet Directory must be established.

If multiple Microsoft Active Directory domains are involved, then the Oracle directory integration and provisioning server synchronizes users and groups from the respective Microsoft Active Directory domains into Oracle Internet Directory. Before the provisioning can take place, a one-way synchronization between Oracle Internet Directory and a domain controller on each Microsoft Active Directory domain must be established.

Passwords are not migrated from Microsoft Active Directory to Oracle Internet Directory.

Configuration of Integration with Microsoft Active Directory

This section contains these topics:

Configuring the Realm

To configure the realm, do the following:

1. Choose the realm DN structure as described in the section "Choose the Structure of the Directory Information Tree" on page 17-6, and, more specifically, in the section "Planning the Deployment" on page 18-9.
2. Select the attribute for the login name of the user. This attribute contains the name of the attribute used for logging in. By default, it is `uid`. If you are integrating with Microsoft Active Directory, and the `userprincipalname` attribute is used for logging in, then you would map `userprincipalname` to the `uid` attribute in Oracle Internet Directory. For more information, see the section "Select the Attribute for the Login Name" on page 17-8.
3. Set up the `usersearchbase` and `groupsearchbase` values in Oracle Internet Directory. These values indicate to the various Oracle components where to look for users and groups in Oracle Internet Directory. They are set to default values during installation. However, in deployments requiring integration with Active Directory, you may need to reset these values so that they correspond to the DIT structures in the two directories. Be sure to set them correctly. Otherwise, even if the synchronization seems to function properly, components still may be unable to access users and groups in Oracle Internet Directory.

   To illustrate how you might configure the user search base and group search base: In the example in Figure 18–3 on page 18-10, the value of `usersearchbase` should be set to `cn=users,dc=us,dc=MyCompany,dc=com` or one of its parents. Similarly, assuming there is a subtree named `groups` in the DIT, the multivalued `groupsearchbase` attribute should be set to both of the following:

   - `cn=groups,dc=us,dc=MyCompany,dc=com` or one of its parents
   - `cn=users,dc=us,dc=MyCompany,dc=com`

   To configure the user search base and group search base, use the Oracle Internet Directory Self-Service Console.
4. Set up the `usercreatebase` and `groupcreatebase` values in Oracle Internet Directory. These values indicate to the various Oracle components where users and groups can be created. They are set to default values during installation.

To illustrate how to configure the user create base and group create base: In the example in Figure 18–3 on page 18-10, the value of `usercreatebase` should be set to `cn=users,dc=us,dc=MyCompany,dc=com` or one of its parents. Similarly, the `groupcreatebase` should be set to `cn=groups,dc=us,dc=MyCompany,dc=com` or one of its parents.

To configure the user create base and group create base, use the Oracle Internet Directory Self-Service Console.

See Also: The section on modifying configuration settings for an identity management realm in *Oracle Identity Management Guide to Delegated Administration*

Configuring Synchronization Profiles

This section describes various customizations that a deployment may require. It contains these topics:

- About the Sample Synchronization Profiles
- Creating Synchronization Profiles
- Configuring the Connection Details for Microsoft Active Directory
- Customizing Mapping Rules
- Customizing the LDAP Schema
- Customizing the Search Filter to Get Information from Microsoft Active Directory
- Synchronizing Deletions from Microsoft Active Directory
- Synchronizing Passwords

Note: Be sure your ORACLE home environment variable is set to the correct value; otherwise, the commands specified in various scenarios do not function properly.

About the Sample Synchronization Profiles

During installation, three sample Active Directory Connector synchronization profiles are provided. You can customize these samples to meet your deployment needs. The sample synchronization profiles are:

- `ActiveImport`—The profile for importing changes from Microsoft Active Directory to Oracle Internet Directory by using the DirSync approach
- `ActiveChgImp`—The profile for importing changes from Microsoft Active Directory to Oracle Internet Directory by using the USN-Changed approach
- `ActiveExport`—The profile for exporting changes from Oracle Internet Directory to Microsoft Active Directory

Whether you use `ActiveImport` or `ActiveChgImp` depends on the method you chose for tracking changes, either DirSync or USN-Changed.

If these sample profiles meet your needs, then copy them and use the exact copies for running Active Directory Connector. If they do not meet your needs, then copy them and customize the copies.

To copy the sample profiles, use the `createprofilelike` (`cpl`) command of the Directory Integration and Provisioning Assistant, then enable the profile by following the instructions in Chapter 7, "Administration of Directory Synchronization". When

you restart the Oracle directory integration and provisioning server, it uses the duplicate profile for synchronization, automatically refreshing its cache with any changed information.

Mapping Rules Mapping rules, an important part of the synchronization profile, determine the directory information to be synchronized and how it is to be transformed when synchronized. You can change mapping rules at run time to meet your requirements.

Each sample Active Directory synchronization profile includes default mapping rules. These rules contain a minimal set of default user and group attributes configured for out-of-the-box synchronization.

Note: When a synchronization is underway, it relies on the mapping rules configured prior to any changes in the directory. To ensure consistent mapping, you may need to remove an already synchronized entry or perform a full synchronization.

See Also:

- "Creating Synchronization Profiles" on page 18-18 for instructions on how to modify the sample profiles to meet your needs
- "Configuring Mapping Rules" on page 6-3 for instructions on how to create mapping rules
- "Customizing Mapping Rules" on page 18-19 for instructions on how to modify the mapping rules to meet your needs

Creating Synchronization Profiles

To create new profiles, copy the sample profiles provided during installation and modify the copies.

To create and configure new profiles, use the Directory Integration and Provisioning Assistant. The Assistant can be invoked as a command-line tool or a graphical interface tool.

- To invoke the Assistant as a command-line tool enter `dipassistant`.
- To invoke the Assistant as a graphical interface tool, enter the following command:

  ```
  $ORACLE_HOME/bin/dipassistant -gui
  ```

  This displays the Oracle Directory Integration and Provisioning Server Administration tool, which provides a subset of the functionality provided through the command-line version of the tool.

  See Also:

  - Chapter 3, "Oracle Directory Integration and Provisioning Administration Tools" for a detailed description of each tool
  - The `dipassistant` section in the Oracle Directory Integration and Provisioning tools chapter of the *Oracle Identity Management User Reference*

Configuring the Connection Details for Microsoft Active Directory

You can configure the Active Directory Connector by using either the Oracle Directory Integration and Provisioning Server Administration tool or the express configuration option of the Directory Integration and Provisioning Assistant. Using either of these, you can specify the connection details as input to the script. This is the recommended method for configuring these details.

You can also create the profiles based on the template properties file provided during installation. If you are doing this, then you must specify the connection details in the `odip.profile.condirurl`, `odip.profile.condiraccount`, and `odip.profile.condirpassword` properties of the profile.

In addition to specifying the connection details, you must also ensure that the user account in Active Directory has the privileges to replicate directory changes for every domain of the forest monitored for changes. You can do this by one of the following methods:

- Grant to this account Domain Administrative permissions
- Make this account a member of the Domain Administrator's group
- Grant to this account Replicating Directory Changes permissions for every domain of the forest that is monitored for changes

To grant this permission to a non-administrative user, follow the instructions in the "More Information" section of the Microsoft Help and Support article "How to Grant the 'Replicating Directory Changes' Permission for the Microsoft Metadirectory Services ADMA Service Account" available at `http://support.microsoft.com/`.

Customizing Mapping Rules

Mapping rules govern the way data is transformed when a source directory and a destination directory are synchronized. Customize the default mapping rules found in the sample profiles when you need to do the following:

- Change distinguished name mappings. The distinguished name mappings establish how the Microsoft Active Directory DIT maps to the Oracle Internet Directory DIT.
- Change the attributes that need to be synchronized.
- Change the transformations (mapping rules) that occur during the synchronization.

You can perform any mapping if the resulting data in the destination directory conforms to the schema in that directory.

Note: For password synchronizations, there are additional mapping considerations. See the section "Synchronizing Passwords" on page 18-24.

See Also: The section "Configuring Mapping Rules" on page 6-3 for a full discussion of mapping rules

Distinguished Name Mapping You can change how the DIT in Active Directory maps to the one in Oracle Internet Directory.

Example 18–1 Example of Distinguished Name Mapping

```
Distinguished Name Rules
%USERBASE INSOURCE%:%USERBASE ATDEST%:
```

USERBASE refers to the container from which Microsoft Active Directory users and groups must be mapped. Usually, this is the users container under the root of the Microsoft Active Directory domain.

Example 18–2 Example of One-to-One Distinguished Name Mapping

For one-to-one mapping to occur, the DN in Microsoft Active Directory must match that in Oracle Internet Directory.

In this example, the DN in Microsoft Active Directory matches the DN in Oracle Internet Directory. More specifically:

- The Microsoft Active Directory host is in the domain us.mycompany.com, and, accordingly, the root of the Microsoft Active Directory domain is us.mycompany.com. A user container under the domain would have a DN value cn=users,dc=us,dc=mycompany,dc=com.
- Oracle Internet Directory has a default realm value of dc=us,dc=mycompany,dc=com. This default realm automatically contains a users container with a DN value cn=users,dc=us,dc=mycompany,dc=com.

Because the DN in Microsoft Active Directory matches the DN in Oracle Internet Directory, one-to-one distinguished name mapping between the directories can occur.

If you plan to synchronize only the cn=users container under dc=us,dc=mycompany,dc=com, then the domain mapping rule is:

```
Distinguished Name Rules
cn=users,dc=us,dc=mycompany,dc=com:cn=users,dc=us,dc=mycompany,dc=com
```

This rule synchronizes every entry under cn=users,dc=us,dc=mycompany,dc=com. However, the type of object synchronized under this container is determined by the attribute-level mapping rules that follow the DN Mapping rules.

If you plan to synchronize the entry cn=groups,dc=us,dc=mycompany,dc=com under cn=users,dc=us,dc=mycompany,dc=com then the domain mapping rule is as follows:

```
cn=groups,dc=us,dc=mycompany,dc=com: cn=users,dc=us,dc=mycompany,dc=com
```

Attribute-Level Mapping Attribute-level mapping specifies:

- The attributes in source directory that are to be synchronized
- The corresponding attributes in the target directory with which they are to be synchronized
- Any transformation of attribute values that is to occur as the data is synchronized from one directory to the other

The following attribute-level mapping is mandatory for all objects:

```
ObjectGUID:  :  : :orclObjectGUID:
ObjectSID:  :  : :orclObjectSID:
```

Example 18–3 Attribute-Level Mapping for the User Object

```
SAMAccountName:1: :user:orclADSAMAccountName: :orclADUser
```

```
userPrincipalName: : :user:orclADUserPrincipalName:
:orclADUser:userPrincipalName
```

Example 18–4 Attribute-Level Mapping for the Group Object

```
SAMAccountName:1: :user:orclADSAMAccountName: :orclADGroup
```

Here, `SAMAccountName` and `userPrincipalName` from Microsoft Active Directory are mapped to `orclADSAMAccountName` and `orclADUserPrincipalName` in Oracle Internet Directory.

Adding another attribute to be synchronized requires adding another rule, as previously indicated earlier. Similarly, if an attribute no longer needs to be synchronized, then the corresponding rule needs to be removed or put in a comment.

See Also:

- The section "Supported Attribute Mapping Rules and Examples" on page 6-8 for examples of how attribute values are transformed when synchronized from one directory to another
- The file $*ORACLE_HOME*/ldap/odi/conf/activeimp.map.master for an example of import mapping rules.

How to Customize the Mapping Rules To customize the mapping rules:

1. Make a duplicate of the sample mapping rules file based on your deployment scenario—for example, whether you are using the DirSync approach or the USN-Changed approach, or whether or not you are doing one-to-one mapping.
2. Edit the sample mapping rules file to make the previously discussed modifications. The sample mapping rules files are stored in the directory $*ORACLE_HOME*/ldap/odi/conf with the extension of `map.master` for the various profiles. You can find instructions for editing mapping rules in "Configuring Mapping Rules" on page 6-3.
3. After the changes are made, enter the following command:

   ```
   $ORACLE_HOME/bin/dipassistant modifyprofile -profile profile_name
   -host oid_host -port oid_port -dn DN -passwd password
   odip.profile.mapfile=path_name
   ```

   For example:

   ```
   $ORACLE_HOME/bin/dipassistant modifyprofile -profile my_profile
   -host my_host -port 3060 -dn cn=orcladmin -passwd welcome1
   odip.profile.mapfile=my_profile.map
   ```

   See Also: The `dipassistant` section in the Oracle Directory Integration and Provisioning tools chapter of the *Oracle Identity Management User Reference*

Customizing the LDAP Schema

Customizing the LDAP schema is required if:

- A directory deployment contains schema extensions such as custom object classes and attributes
- The custom attributes must be synchronized from one directory server to the other

To customize the LDAP schema, you must:

- Identify the schema extensions on the source directory
- Create those extensions on the target directory before starting the data migration and the synchronization.

Note: In addition to creating schema extensions, you must also add the attribute to be synchronized with the corresponding object classes to the mapping rules.

See Also:

- The chapter on administering the schema in *Oracle Internet Directory Administrator's Guide* for instructions on customizing the schema in Oracle Internet Directory
- Microsoft documentation available at `http://msdn.microsoft.com/` for instructions on customizing the schema in Microsoft Active Directory

Customizing the Search Filter to Get Information from Microsoft Active Directory

By default, Active Directory Connector retrieves changes to all objects in the container configured for synchronization. If you are interested in retrieving only a certain type of change, for example only changes to users and groups, then you should configure an LDAP search filter. This filter screens out changes that are not required when Active Directory Connector queries Active Directory. The filter is stored in the `searchfilter` attribute in the synchronization profile.

In the sample profiles `activeChgImp` and `activeImport`, only groups and users are retrieved from Microsoft Active Directory. Computers are not retrieved. The value of the `searchfilter` attribute is set as:
`searchfilter=(|(objectclass=group)(&(objectclass=user)(!(objectclass=computer))).`

You can use either Oracle Directory Integration and Provisioning Server Administration tool or Directory Integration and Provisioning Assistant to update the `searchfilter` attribute.

To customize the search filter by using the Directory Integration and Provisioning Assistant:

1. Enter the following command to customize the Connected Directory Matching Filter (`orclODIPConDirMatchingFilter`) attribute:

   ```
   $ORACLE_HOME/bin/dipassistant modifyprofile -D bindDn -w password -profile
   profName odip.profile.condirfilter=searchfilter=(|(objectclass=group)
   (objectclass=organizationalunit)(&(objectclass=user)(!(objectclass=computer))))
   ```

2. Enter the following command to customize the OID Matching Filter (`orclODIPOIDMatchingFilter`) attribute:

   ```
   $ORACLE_HOME/bin/dipassistant modifyprofile -D bindDn -w password
   -profile profName odip.profile.oidfilter=orclObjectGUID
   ```

To customize the search filter by using the Oracle Directory Integration and Provisioning Server Administration tool:

1. Launch the Oracle Directory Integration and Provisioning Server Administration tool by entering:

```
$ORACLE_HOME/bin/dipassistant -gui
```

2. In the navigator pane, expand ***directory_integration_and_provisioning_server***, then expand **Integration Profile Configuration**.
3. Select the configuration set, and, in the right pane, select the profile you want to customize. The Integration Profile window appears.
4. In the Integration Profile window, select the Mapping tab. The fields in this tab page are described in "Mapping" on page A-8.
5. In the Mapping tab page, in the Connected Directory Matching Filter (`orclODIPConDirMatchingFilter`) and the OID Matching Filter (`orclODIPOIDMatchingFilter`) fields, enter the appropriate values for the `searchfilter` attribute. Instructions for specifying the `searchfilter` attribute are provided in the section "Filtering Changes with an LDAP Search" on page 6-13.
6. Choose **OK**.

Note: All attributes specified in the `searchfilter` attribute should be configured as indexed attributes in Microsoft Active Directory.

See Also: The appendix on the LDAP filter definition in *Oracle Internet Directory Administrator's Guide* for instructions on configuring an LDAP search filter

Synchronizing Deletions from Microsoft Active Directory

Active Directory deletions can be synchronized with Oracle Internet Directory by querying for them in Active Directory. The way to do this depends on whether you are using the DirSync approach or the USN-Changed approach.

For the DirSync approach, the Active Directory user account that the Oracle directory integration and provisioning server uses to access Active Directory must have Domain Administrative permissions, belong to the Domain Administrators group, or be explicitly granted Replicating Directory Changes permissions.

See Also: Article ID 303972 at `http://support.microsoft.com` for information on how to grant Replicating Directory Changes permissions

For the USN-Changed approach, the Active Directory user account that the Oracle directory integration and provisioning server uses to access Active Directory must have "List Content" and "Read Properties" permission to the `cn=Deleted Objects` container of a given domain. In order to set these permissions, you must use the dsacls.exe command that is available with recent versions of Active Directory Application Mode (ADAM). You can download the most recent version of ADAM at `http://www.microsoft.com/downloads/`.

Regardless of whether you are using the DirSync approach or the USN-Changed approach to synchronize deletions in Active Directory with Oracle Internet Directory, if you create a matching filter for the `ActiveImport` profile (for the DirSync approach) or the `ActiveChgImp` profile (for the USN-Changed profile) be sure to include only the following key Active Directory attributes:

- `Object-GUID`
- `Object-SID`

- `Object-Dist-Name`
- `USN`

In you specify any attributes in a matching filter other than the preceding key attributes, deletions in Active Directory are not propagated to Oracle Internet Directory.

See Also:

- Article ID 230113 at `http://support.microsoft.com` for more information on deleting items from Active Directory
- The attribute reference chapter in the *Oracle Identity Management User Reference* for a listing of the standard LDAP attributes that Oracle Internet Directory supports

Synchronizing Passwords

You can synchronize Oracle Internet Directory passwords with Active Directory. You can also make passwords stored in Microsoft Active Directory available in Oracle Internet Directory. Password synchronization is possible only when the directories run in SSL mode 2, that is, server-only authentication.

Synchronizing Passwords from Oracle Internet Directory to Microsoft Active Directory Before Active Directory Connector can synchronize passwords in this direction, do the following:

- Add a mapping rule that enables password synchronization. For example:

```
Userpassword: : :inetorgperson:unicodepwd: :user
```

- Enable the password policy and reversible password encryption in the Oracle directory server. To do this, assign a value of `1` to the `orclPwdPolicyEnable` and `orclpwdEncryptionEnable` attributes in the entry `cn=PwdPolicyEntry,cn=common,cn=products,cn=oraclecontext,`*`DN_of_realm`*. You can do this by using either Oracle Directory Manager or ldapmodify.

See Also:

- "Configuring the Active Directory Connector for Synchronization in SSL Mode" on page 18-26
- The section "Configuring Mapping Rules" on page 6-3 for instructions on adding mapping rules
- The chapter on directory storage of password verifiers in *Oracle Internet Directory Administrator's Guide* for information about enabling reversible encryption

Synchronizing from Microsoft Active Directory to Oracle Internet Directory Because passwords in Microsoft Active Directory cannot be accessed by LDAP clients, you cannot synchronize Oracle Internet Directory passwords with Microsoft Active Directory in Oracle Application Server. However, if a deployment requires passwords to be available in Oracle Internet Directory, then Oracle recommends the following two methods:

- Build a custom plug-in for Active Directory that captures a password change and synchronizes it with Oracle Internet Directory. For more information:

  - See the chapter about the Oracle Internet Directory plug-in framework in *Oracle Internet Directory Administrator's Guide*
  - Visit the Microsoft Developer Network (MSDN) at `http://msdn.microsoft.com/`

- Manage Active Directory passwords from the Oracle environment. With this method, passwords are available in both Oracle Internet Directory and Microsoft Active Directory. The Active Directory Connector can synchronize the two directories.

  Note: To synchronize passwords, you must enable SSL mode as discussed in "Configuring the Active Directory Connector for Synchronization in SSL Mode" on page 18-26.

Customizing Access Control Lists

This section discusses how to customize ACLs for import profiles, export profiles, and for other Oracle components. It contains these topics:

- Customizing ACLs for Import Profiles
- Customizing ACLs for Export Profiles
- ACLs for Other Oracle Components

Customizing ACLs for Import Profiles

The import profile is the identity used by the Oracle directory integration and provisioning server to access Oracle Internet Directory. ACLs must enable the import profile to add, modify, and delete objects in either the users and groups containers or the subtree where entries are accessed. By default, import profiles are part of the Realm Administrators group (`cn=RealmAdministrators, cn=groups,cn=OracleContext,`*`realm_DN`*) in the default realm. This group grants privileges to perform all operations on any entry under the DN of the default realm.

You should not need to customize the ACLs for import synchronization with the default realm that is installed with Oracle Internet Directory Release 10*g* Release 2 (10.1.2). If you are upgrading from an earlier version of Oracle Internet Directory, or if the synchronization is with a nondefault Oracle Internet Directory realm, then be sure that the necessary privileges in the proper subtree or containers are granted to the import profiles handling the synchronization.

For an ACL template in LDIF format, see the file $*ORACLE_HOME*/ldap/schema/oid/oidRealmAdminACL.sbs. If you have not changed the ACLs on the default realm, then this template file can be applied directly after instantiating the substitution variables, replacing `%s_SubscriberDN%` with the default realm DN in Oracle Internet Directory and replacing `%s_OracleContextDN%` with `cn=OracleContext,`*`default_realm_DN`* respectively. For example, if `realmacl.ldif` is the instantiated file, then you can upload it by using the following ldapmodify command:

```
$ORACLE_HOME/bin/ldapmodify -h <OID host> -p <OID port>
-D "DN of privileged OID user" -w "password of privileged OID user"
-v -f realmacl.ldif
```

See Also: The chapter on access controls in *Oracle Internet Directory Administrator's Guide*

Customizing ACLs for Export Profiles

To enable the Oracle directory integration and provisioning server to access Active Directory, you must create an identity in Active Directory. This identity is configured in each export profile.

ACLs for Other Oracle Components

Default ACLs enable you to create, modify, and delete users and groups, but only in the users and groups containers under the default realm. To synchronize objects in other containers, you must customize the ACLs.

There are sample ACL files that you can use to customize ACLs for Oracle Components. These sample files are installed in the directory $*ORACLE_HOME*/ldap/schema/oid/. They are:

- `oidUserAdminACL.sbs`—Grants necessary rights to the subtree for Oracle components to manage and access users
- `oidGroupAdminACL.sbs`—Grants necessary rights to the subtree for Oracle components to manage and access groups.
- `oidUserAndGroupAdminACL.sbs`—Grants the privileges for Oracle components to manage and access users and groups in the subtree.

You can customize your ACL policy to grant privileges on a container-by-container basis with the required rights.

> **See Also:** The chapter on access control in *Oracle Internet Directory Administrator's Guide* for instructions on customizing ACLs

Configuring the Active Directory Connector for Synchronization in SSL Mode

Active Directory Connector uses SSL to secure the synchronization process. Whether or not you synchronize in the SSL mode depends on your deployment requirements. For example, synchronizing public data does not require SSL, but synchronizing sensitive information such as passwords does. To synchronize password changes between Oracle Internet Directory and Microsoft Active Directory, you must use SSL mode with server-only authentication, that is, SSL Mode 2.

Securing the channel requires:

- Enabling SSL between Oracle Internet Directory and the Oracle directory integration and provisioning server
- Enabling SSL between the Oracle directory integration and provisioning server and Microsoft Active Directory

Although you can enable SSL either between Oracle Internet Directory and the Oracle directory integration and provisioning server or between that server and Microsoft Active Directory, Oracle recommends that you completely secure the channel before you synchronize sensitive information. In certain cases, such as password synchronization, synchronization can occur only over SSL.

Configuring SSL requires the following:

- Running the Oracle directory server in SSL mode as described in the chapter on Secure Sockets Layer (SSL) in *Oracle Internet Directory Administrator's Guide*
- Running the Oracle directory integration and provisioning server in the SSL mode as described in Chapter 2, "Security Features in Oracle Directory Integration and Provisioning". The SSL mode should be the same as the one in which Oracle Internet Directory server was started. When starting the Oracle directory

integration and provisioning server, specify the `sslauth` parameter to `1` for no authentication or `2` for server-only authentication.

- Running the Microsoft Active Directory server in SSL mode. Communication with Microsoft Active Directory over SSL requires SSL Mode 2, that is, server-only authentication. This requires that both Oracle Internet Directory and the Oracle directory integration and provisioning server be run in SSL mode 2.
- Configuration of the Microsoft Active Directory connector to use SSL. This includes creating a wallet which will contain the certificates for both Oracle Internet Directory and Microsoft Active Directory. For more information, see "Managing the SSL Certificates of Oracle Internet Directory and Connected Directories" on page 4-7.

Note: The Oracle Directory Integration Platform does not support SSL in the client/server authentication mode.

Considerations for Synchronizing with a Multiple-Domain Microsoft Active Directory Environment

This section describes how to import from Microsoft Active Directory to Oracle Internet Directory and export from Oracle Internet Directory to Microsoft Active Directory.

Configuration Required for Importing from Microsoft Active Directory to Oracle Internet Directory

Normally, importing requires configuring one import profile for each Microsoft Active Directory domain regardless of whether you are using the DirSynch approach or the USN-Changed approach. However, if you are using the USN-Changed approach, you can use the Global Catalog to import from an entire Microsoft Active Directory forest. Although this requires configuring only one import profile, consider the following:

- Because Global Catalog is read-only, you can use it only for importing data into Oracle Internet Directory.
- Global Catalog does not contain all the attributes, although the available attributes can be configured in Microsoft Active Directory.
- Because Global Catalog is a global synchronization point, the process can become congested as a result of additional access to the import file.

See Also: The Microsoft Knowledge Base Article 256938 available from Microsoft Help and Support at `http://support.microsoft.com/` for information about Global Catalog attributes in the Microsoft Active Directory schema

Configuration Required for Exporting from Oracle Internet Directory to Microsoft Active Directory

To integrate with multiple-domain Microsoft Active Directory environments, the Oracle directory integration and provisioning server obtains configuration information from each Active Directory domain. You must configure as many export profiles as there are Microsoft Active Directory domains.

Configuring the Active Directory Connector Profiles

The Oracle directory integration and provisioning server includes an express configuration option that you can run with either the Directory Integration and Provisioning Assistant or the Oracle Directory Integration and Provisioning Server Administration tool. Express configuration creates two synchronization profiles, one for import and one for export, using predefined assumptions. After you enable the profiles, you can immediately begin synchronizing users and groups between `cn=users,default_naming_context` in Microsoft Active Directory and `cn=users,default_realm` in Oracle Internet Directory.

The Active Directory connector import and export synchronization profiles created with express configuration are only intended as a starting point for you to use when deploying your integration of Oracle Internet Directory and Microsoft Active Directory. Because the default synchronization profiles are created using predefined assumptions, you must further customize them for your environment.

Note: While customizing the synchronization profiles for your environment, you may need to add test users and groups to facilitate your deployment effort. Be sure to remove any test users and groups when your are finished customizing and testing your synchronization profiles.

WARNING: In order to successfully customize your import and export synchronization profiles, do not enable SSL until you have finished with all other configuration tasks.

In order to successfully complete configuration of the profiles for your environment, be sure to perform the procedures listed in this section in the following order:

1. Preparing for Synchronization
2. Creating Synchronization Profiles with Express Configuration
3. Customizing Attribute Mapping
4. Final Configuration Requirements
5. Configuring Synchronization Profiles for SSL
6. Additional Considerations

Preparing for Synchronization

To prepare for synchronization between Oracle Internet Directory and Microsoft Active Directory:

1. Plan your deployment by reading the following:
   - Chapter 17, "Considerations for Integrating with Third-Party Directories"
   - "Concepts and Architecture of Microsoft Active Directory Integration" on page 18-2
2. Use Oracle Enterprise Manager 10*g* Application Server Control Console to verify that Oracle Internet Directory is running.

See Also:

- *Oracle Internet Directory Administrator's Guide* for information on how to work with the Oracle Enterprise Manager 10*g* Application Server Control Console
- Your Microsoft Active Directory documentation for instructions on how to verify that Microsoft Active Directory is running

3. Create a user account in Microsoft Active Directory with sufficient privileges to perform both import and export operations. Oracle Directory Integration and Provisioning will use this account to log in to Microsoft Active Directory.

   - **For Import Operations from Microsoft Active Directory**: Grant the user account read access privileges to the subtree root. The user account must be able to read all objects under the source container (subtree root) in Active Directory that are to be synchronized with the Oracle directory integration and provisioning server. To verify whether an Active Directory user account has the necessary privileges to all Active Directory objects to be synchronized with Oracle Internet Directory, use the command-line `ldapsearch` utility to perform a subtree search, as follows:

     ```
     $ORACLE_HOME/bin/ldapsearch -h <AD host> -p <AD port> -b "DN of subtree"
     -s sub -D "DN of privileged AD user" -w "password for privileged AD user"
     "objectclass=*"
     ```

     The return results from the `ldapsearch` utility should include all objects of interest, including all attributes and values that will be synchronized.

     To synchronize deletions of users in Active Directory with Oracle Internet Directory, you must grant the user account the necessary privileges by following the instructions in "Synchronizing Deletions from Microsoft Active Directory" on page 18-23.

   - **For Export Operations to Microsoft Active Directory**: Grant the user account the following privileges to the subtree root that is the parent of all the containers to which the Oracle directory integration and provisioning server will export users:
     - Write
     - Create all child objects
     - Delete all child objects

See Also: Your Microsoft Active Directory documentation for information how to grant privileges to user accounts

Creating Synchronization Profiles with Express Configuration

This section describes how to create and customize synchronization profiles with express configuration. It contains these topics:

- Understanding Express Configuration
- Running Express Configuration
- Additional Synchronization Considerations

Understanding Express Configuration To simplify the configuration, the express configuration option assumes the following:

- Only creation and modifications of organizational units, users, and groups are synchronized.

  Entries for Users and groups in Active Directory are located in the container `cn=users,default_naming_context`.

- Entries for users of the default realm in Oracle Internet Directory are located in the container `cn=users,default_realm_DN`.
- Entries for groups of the default realm in Oracle Internet Directory are located in the container `cn=groups,default_realm_DN`
- The method used for tracking changes in Active Directory is the USN-Changed approach.
- The default Active Directory Connector profiles—namely, `ActiveImport`, `ActiveExport`, and `ActiveChgImp`—are present in the Oracle directory server.
- The Directory Integration and Provisioning master mapping rules files created during installation are present in $*ORACLE_HOME*/ldap/odi/conf. The file names are `activechg.map.master` and `activeexp.map.master`.
- The logon credential is that of a Directory Integration and Provisioning administrator with sufficient privileges to configure a profile, a realm, and access controls on the Users container in the Oracle directory server. Members of the Directory Integration and Provisioning Administrators group (`cn=dipadmingrp,cn=odi,cn=oracle internet directory`) have the necessary privileges.
- Connections to Active Directory or Oracle Internet Directory are NOT over SSL.

Perform the following steps to run express configuration and verify that users and groups are synchronizing between `cn=users,default_naming_context` in Microsoft Active Directory and `cn=users,default_realm` in Oracle Internet Directory:

1. Run express configuration by following the procedures described in "Running Express Configuration" on page 18-31.
2. Enable the import and export synchronization profiles by using either the Oracle Directory Integration and Provisioning Server Administration tool or the Directory Integration and Provisioning Assistant with the `modifyprofile` option. For example, the following Directory Integration and Provisioning Assistant command enables an import profile named `myprofile`:

   ```
   $ORACLE_HOME/bin/dipassistant modifyprofile -host myhost -port 3060
   -passwd my_password -file import.profile -dn bind_DN
   -passwd Password_of_bind_DN -profile myprofile odip.profile.status=ENABLE
   ```

3. Start the Oracle directory integration and provisioning server by following the instructions described in "Starting, Stopping, and Restarting the Oracle Directory Integration and Provisioning Server" on page 4-8.
4. Wait until the scheduling interval has elapsed and verify that synchronization has started by entering the following command:

   ```
   $ORACLE_HOME/bin/ldapsearch -h <OID host> -p <OID port>
   -D "DN of privileged OID user" -w "password of privileged OID user"
   -b "orclodipagentname=activechgimp,cn=subscriber profile,cn=changelog
   subscriber,cn=oracle internet directory" -s base "objectclass=*"
   ```

```
orclodipsynchronizationstatus orclodiplastsuccessfulexecutiontime
```

> **Note:** The default scheduling interval is 60 seconds (1 minute). You can use the Directory Integration and Provisioning Assistant or the Oracle Directory Integration and Provisioning Server Administration tool to change the default scheduling interval. For more information, see Chapter 3, "Oracle Directory Integration and Provisioning Administration Tools".

When synchronization is successfully started:

- The value of the Synchronization Status attribute is `Synchronization Successful`.
- The value of the Last Successful Execution Time attribute is the specific date and time of that execution. Note that this must be close to the current date and time.

An example of a result indicating successful synchronization is:

```
Synchronization successful November 04, 2003 15:56:03
```

> **Note:**
>
> - The date and time must be close to current date and time.
> - When running the ldapsearch command, you need the dipadmin password, which, as established at installation, is the same as orcladmin password.

5. After verifying that synchronization has started, examine the entries in Oracle Internet Directory and Microsoft Active Directory to confirm that users and groups are synchronizing between `cn=users,default_naming_context` in Microsoft Active Directory and `cn=users,`*`default_realm`* in Oracle Internet Directory.

Running Express Configuration You can run express configuration using either the Oracle Directory Integration and Provisioning Server Administration or the Directory Integration and Provisioning Assistant, as described in the following sections:

- Running Express Configuration with the Oracle Directory Integration and Provisioning Server Administration Tool
- Running Express Configuration with the Directory Integration and Provisioning Assistant

Running Express Configuration with the Oracle Directory Integration and Provisioning Server Administration Tool

To perform an express configuration of the Active Directory Connector:

1. Launch the Oracle Directory Integration and Provisioning Server Administration tool by entering:

   ```
   $ORACLE_HOME/bin/dipassistant -gui
   ```

2. In the Oracle Directory Integration and Provisioning Server Administration tool, expand *directory_server*, then Integration Profile Configuration, and select Active

Directory Connector Configuration. The corresponding tab pages appear in the right pane.

3. In the Active Directory Connector Express Synchronization tab page, enter the appropriate values.
4. Choose Apply.

Running Express Configuration with the Directory Integration and Provisioning Assistant

To perform an express configuration of the Active Directory Connector:

1. Launch the Directory Integration and Provisioning Express Configuration Tool:

```
$ORACLE_HOME/bin/dipassistant expressconfig
[-h oracle_internet_directory_host
-p oracle_internet_directory_port -configset configuration_set_entry]
```

The arguments in the preceding example are listed in Table 18–4.

Table 18–4 Arguments for the Directory Integration and Provisioning Express Configuration Tool

Argument	Description
oracle_internet_directory_host	Host of the Oracle directory server. The default is the local host.
oracle_internet_directory_port	Non-SSL port for Oracle Internet Directory. The default is `389`.
configuration_set_entry	Configuration set for Oracle Directory Integration and Provisioning. The default is `1`.

2. When prompted, enter the following information:
   - Oracle Internet Directory credentials. You must specify the super user, that is, `cn=orcladmin`, or any user that is a member of the Directory Integration and Provisioning Administrators group (`cn=dipadmingrp,cn=odi,cn=oracle internet directory`).
   - Active Directory connection details and credentials of a privileged user. To synchronize deletions, you must have the necessary administrative privileges in Microsoft Active Directory, for example `administrator@MyCompany.com` if the host on which Microsoft Active Directory is installed is *hostname*`@us.oracle.com`.
   - Name to identify the synchronization profiles to be created. For example, if you specify the name `abc`, then the tool creates two profiles: `abcImport` and `abcExport`.
   - (Optional) Appropriate ACLs on the `cn=users` container. You can choose to enable users and groups to be managed by Oracle components under the `cn=users` container. If you customize ACLs in this way, then the original ACLs are saved in $*ORACLE_HOME*/ldap/odi/archive/profile_name_prefix_useracl.ldif.

Additional Synchronization Considerations This section describes additional issues that you may need to consider when configuring your synchronization profiles. It contains these topics:

- Handling Synchronization Errors

- Synchronizing Deletions in Active Directory
- Using DirSync Change Tracking for Import Operations

Handling Synchronization Errors

While examining synchronization results, you may notice that the Oracle directory integration and provisioning server is attempting to repeatedly process the same change. This indicates that an error is occurring during synchronization of that change. By default, the Oracle directory integration and provisioning server will continue processing a change until the error is resolved. However, you can configure the Oracle directory integration and provisioning server to skip any changes that cause an error. For more information, see "The SkipErrorToSyncNextChange Parameter" on page 6-3.

> **See Also:** Appendix C, "Troubleshooting Oracle Directory Integration and Provisioning"

Synchronizing Deletions in Active Directory

In order to synchronize deletions in Active Directory with Oracle Internet Directory, you must grant the necessary privilege to the Active Directory user account that the Oracle directory integration and provisioning server uses to perform synchronizations with Active Directory. For more information, see "Synchronizing Deletions from Microsoft Active Directory" on page 18-23.

Using DirSync Change Tracking for Import Operations

The import synchronization profile created with express configuration uses the USN-Changed approach for tracking changes. To modify the import synchronization profile so it uses the DirSync change tracking approach:

> **Note:** You may want to backup your current import synchronization profile before performing the following procedures. You can create a backup copy of a profile by using the Directory Integration and Provisioning Assistant's `createprofilelike` command. For more information, see the `dipassistant` section in the Oracle Directory Integration and Provisioning tools chapter of the *Oracle Identity Management User Reference*.

1. You can use the activeimp.cfg.master file, located in your $*ORACLE_HOME*/ldap/odi/conf directory, to change the import synchronization profile from the USN-Changed approach to DirSync. Use the following command to update the profile:

   ```
   $ORACLE_HOME/bin/dipassistant modifyprofile -profile profile_name
   odip.profile.configfile=$ORACLE_HOME/ldap/odi/conf/activeimp.cfg.master
   ```

2. Update the last change number by running the following command:

   ```
   $ORACLE_HOME/bin/dipassistant modifyprofile -profile profile_name -updcln
   ```

   In order to update the last change number, the value assigned to the `odip.profile.condirurl` property in the import synchronization profile must be for a non-SSL connection. If you have already configured the import synchronization profile for SSL, then before attempting to update the last change number, you must temporarily change the value assigned to the `odip.profile.condirurl` property so it points to a non-SSL port.

See Also: "Configuring the Connection Details for Microsoft Active Directory" on page 18-19

Customizing Attribute Mapping

Once you have established a working synchronization between Oracle Internet Directory and Microsoft Active Directory, you can customize the attribute mapping rules for your synchronization profiles to meet the needs of your deployment. To customize the attribute mapping rules for your synchronization profiles:

1. When you use express configuration to create import and export synchronization profiles, mapping files are created for each profile in the $*ORACLE_HOME*/ldap/conf directory. The mapping files are named *profile_name*Import.map and *profile_name*Export.map. For example, if you enter "abc" when express configuration prompts you for the name of your profile, your import mapping files will be named abcImport.map and abcExport.map. Modify the mapping rules in your mapping files as needed by following the instructions described in "Customizing Mapping Rules" on page 18-19.
2. Wait until the scheduling interval has elapsed, and then check the synchronized users and groups to ensure that the attribute mapping rules meet your requirements.
3. Repeat Step 1 through Step 2 until the synchronized users and groups contain the attributes you need.

Tip: You may find it helpful to add test users and groups to Oracle Internet Directory or Microsoft Active Directory when customizing attribute mapping rules.

Final Configuration Requirements

This section describes the final configuration requirements for the import and export synchronization profiles created with express configuration. It contains these topics:

- Customizing DN Mapping Rules
- Synchronizing Multiple Domains
- Performing Initial Bootstrapping
- Granting Privileges to Non-Default Realms

Customizing DN Mapping Rules Once you have finished customizing the attribute mapping rules for your synchronization between Oracle Internet Directory and Microsoft Active Directory, you should customize the DN mapping rules for your synchronization profiles to meet the needs of your deployment.

WARNING: If you do not correctly map DN rules, then configuring multiple Microsoft Active Directory domains against a single instance of Oracle Internet Directory can result in name collision. This is because the container `cn=users,default_naming_context` **in each of the multiple domains in Microsoft Active Directory is synchronized to the same container,** `cn=users,default_realm`**, in Oracle Internet Directory.**

To customize the DN mapping rules for your synchronization profiles:

1. Modify the DN mapping rules in your mapping files as needed by following the instructions described in "Customizing Mapping Rules" on page 18-19.
2. Wait until the scheduling interval has elapsed, and then check the synchronized users and groups to ensure that the DN mapping rules meet your requirements.
3. Repeat Step 1 through Step 2 until the DN mapping rules meet the needs of your deployment.

> **Tip:** You may find it helpful to add test users and groups to Oracle Internet Directory or Microsoft Active Directory when customizing DN mapping rules.

Synchronizing Multiple Domains When synchronizing with multiple Active Directory domains, you need separate import and export synchronization profiles for each domain in most cases. However, the profiles for each domain should be very similar. The only exception involves using Global Catalog with import synchronization profiles. In this case, you only need to create a single import synchronization profile for the entire Active Directory forest. For more information, see "Configuration Required for Importing from Microsoft Active Directory to Oracle Internet Directory" on page 18-27.

> **Note:** Be sure to perform attribute and DN mapping before attempting to synchronize with multiple domains.

The best approach to creating separate import and export synchronization profiles for multiple domains is as follows:

1. Customize the import and export synchronization profiles for a single domain, using the procedures described earlier in this section.
2. Once you have finished customizing the import and export synchronization profiles for the first domain, use the Directory Integration and Provisioning Assistant's `createprofilelike` command to duplicate profiles, as follows.

```
$ORACLE_HOME/bin/dipassistant createprofilelike [-h hostName] [-p port]
[-D bindDn] [-w password] -profile origProfName -newprofile newProfName
```

3. Use the Directory Integration and Provisioning Assistant's `modifyprofile` command to customize the profiles for each additional Active Directory domain, as follows:

```
$ORACLE_HOME/bin/dipassistant modifyprofile [-h hostName] [-p port]
[-D bindDn] [-w password] {-f fileName | -profile profName [-updlcn] }
[propName1=value] [propName2=value]...
```

4. If necessary, update the connection details for each domain by following the instructions listed in "Configuring the Connection Details for Microsoft Active Directory" on page 18-19.
5. Update the last change number in the import and export synchronization profiles for each domain by running the following command:

```
$ORACLE_HOME/bin/dipassistant modifyprofile -profile profile_name -updcln
```

In order to update the last change number, the value assigned to the `odip.profile.condirurl` property in the import synchronization profile must be for a non-SSL connection. If you have already configured the import synchronization profile for SSL, then before attempting to update the last change

number, you must temporarily change the value assigned to the `odip.profile.condirurl` property so it points to a non-SSL port.

6. Repeat Steps 2 through 5 for each Active Directory domain to which you need to synchronize.

Performing Initial Bootstrapping Once you have finished configuring your import and export synchronization profiles, including customizing attribute mappings, DN mappings, and configuring for multiple Active Directory realms, you can migrate data from an Active Directory domain to Oracle Internet Directory by using the `bootstrap` option of the Directory Integration and Provisioning Assistant. This is described in "Bootstrapping Data Between Directories" on page 18-52.

Granting Privileges to Non-Default Realms If you need to synchronize Microsoft Active Directory with an Oracle Internet Directory subtree that is not in the default realm, then be sure to grant the necessary privileges to the import and export synchronization profiles. The import synchronization profile must have privileges to create, modify, and delete entries while the export synchronization profile must have read privileges to Oracle Internet Directory, including `cn=changelog`.

Configuring Synchronization Profiles for SSL

Your last step in customizing the import and export synchronization profiles should be to enable SSL. By default, SSL is not enabled for the import and export synchronization profiles created with express configuration. This section describes how to enable SSL for Active Directory synchronizations.

> **Note:** Be sure that you can successfully synchronize users in non-SSL mode before attempting to configure your synchronization profiles for SSL.

1. Follow the instructions in "Configuring the Active Directory Connector for Synchronization in SSL Mode" on page 18-26.
2. Once SSL is enabled for Active Directory and Oracle Internet Directory, you can modify the Active Directory connection information, including the host name and profile, using the Directory Integration and Provisioning Assistant's `modifyprofile` command, as follows:

```
$ORACLE_HOME/bin/dipassistant modifyprofile <-h hostName> <-p port>
-profile profilename odip.profile.condirurl= ad_host_name:636:1
```

3. Restart the Oracle directory integration and provisioning server by following the instructions "Starting, Stopping, and Restarting the Oracle Directory Integration and Provisioning Server" on page 4-8.
4. Add a test user and verify that it synchronizes successfully. If the test user does not synchronize successfully, then troubleshoot your SSL configuration.

Additional Considerations

Read the following topics for additional configuration requirements:

- "Configuring the Realm" on page 18-19
- "Configuring the Active Directory External Authentication Plug-in" on page 18-37
- "Configuring Windows Native Authentication" on page 18-39

- "ACLs for Other Oracle Components" on page 18-26

Configuring the Active Directory External Authentication Plug-in

This section explains how to delete, disable, and reenable the Active Directory external authentication plug-in. It contains these topics:

- Installing Active Directory External Authentication Plug-ins
- Installing Active Directory External Authentication Plug-ins for Multiple Domains
- Enabling the Active Directory External Authentication Plug-ins
- Testing the Active Directory External Authentication Plug-ins

Installing Active Directory External Authentication Plug-ins

To install the plug-in:

1. Execute the oidspadi.sh script by entering:

```
cd $ORACLE_HOME/ldap/admin
sh oidspadi.sh
```

> **Note:** To run shell script tools on the Windows operating system, you need one of the following UNIX emulation utilities:
>
> - Cygwin 1.3.2.2-1 or later. Visit `http://sources.redhat.com/`
> - MKS Toolkit 6.1. Visit `http://www.datafocus.com/`

   If you are using the Windows operating system, then execute oidspadi.sh after you have installed the UNIX emulation utility by entering:

```
sh oidspadi.sh
```

2. Enter the Microsoft Active Directory host name. This is the Microsoft Active Directory with which you are going to synchronize. This value is required.
3. Specify whether to use an SSL connection to Microsoft Active Directory. If you choose to use SSL, then you need to enter the following:
   - The Microsoft Active Directory SSL connection port number
   - The location of the Oracle wallet. This wallet needs to have the valid certificate from the Microsoft Active Directory that you are trying to connect to.
   - The Oracle wallet password.

     When specifying the wallet location on the Microsoft Windows operating system, add an additional backslashes (\). For example, if the wallet location is `D:storage\wallet`, then enter `D:\\storage\\wallet`.
4. Enter the connect string for the database designated for Oracle Internet Directory.
5. Enter the ODS password for Oracle Internet Directory
6. Enter the directory server host name. This value is required.
7. Enter directory server port number. The default port is 389.
8. Enter the password of the Oracle administrator (`orcladmin`). This value is required.

9. (Optional) Enter the distinguished name of the container to which the plug-in needs to be applied. Every entry in this container will be authenticated against Active Directory. Note that this need not necessarily be the User Search Base supplied by using the Oracle Internet Directory Self-Service Console. All the users under this search base are authenticated externally to the Active Directory. If more than one container is specified, then separate the DNs with semi-colons (;).
10. Enter the Plug-in Request Group DN. For security reasons, the plug-in can be invoked only by users belonging to this group. For example, suppose that the Oracle Application Server Single Sign-On administrators are in the group `cn=OracleUserSecurityAdmins,cn=Groups,cn=OracleContext`. If you enter this DN as the value for the Plug-in Request Group DN, then only requests fromOracle Application Server Single Sign-On administrators can trigger the external authentication plug-in. You can enter multiple DN values. Use a semicolon (;) to separate them. This value is not required, but, for security purposes, it should be specified.
11. (Optional) Enter the value of the entry that is to be excluded from authentication to Microsoft Active Directory. This value is the exception to Step 9. You need to enter the value in the standard ldapsearch filter format. For example, if you specify the value `(&(objectclass=inetorgperson)(cn=orcladmin))`, then any entry under the user container specified in Step 9 that has the `cn=orcladmin` and `objectclass=inetorgperson` attribute values will not be authenticated to Microsoft Active Directory.
12. (Optional) Specify the backup Microsoft Active Directory domain controller details.

Installing Active Directory External Authentication Plug-ins for Multiple Domains

You should use a single instance of Global Catalog to configure multiple Active Directory domains for external authentication in Oracle Internet Directory. However, if you cannot configure a single instance of Global Catalog for multiple Active Directory domains in your deployment environment, then install multiple Active Directory external authentication plug-ins for each domain as follows.

1. Copy and edit the Active Directory external authentication plug-in SQL package:
    a. Copy the $*ORACLE_HOME*/ldap/admin/oidspada.pls file to oidspada2.pls, or another unique file name that represents an additional Active Directory domain.
    b. Edit oidspada2.pls (or whatever file name you chose) and replace all five occurrences of "OIDADPSWD" with "OIDADPSW2".
    c. Save and close oidspada2.pls.
2. Copy and edit the Active Directory external authentication plug-in installation script:
    a. Copy the $*ORACLE_HOME*/ldap/admin/oidspadi.sh file to oidspadi2.sh, or another unique file name that represents an additional Active Directory domain.
    b. Edit oidspadi2.sh (or whatever file name you chose) and make the following edits.
    c. Go to line 361 and replace "oidspada.pls" with "oidspad2.pls".
    d. Go to line 380 and replace "cn=adwhencompare" with "cn=adwhencompare2".
    e. Go to line 383 and replace "OIDADPSWD" with "OIDADPSW2".

f. Go to line 390 and replace "adwhencompare" with "adwhencompare2".

g. Go to line 396 and replace "cn=adwhenbind" with "cn=adwhenbind2".

h. Go to line 399 and replace "OIDADPSWD" with "OIDADPSW2".

i. Go to line 406 and replace "adwhenbind" with "adwhenbind2".

j. Save and close oidspadi2.sh.

3. Execute the oidspadi2.sh script by following the instructions in "Installing Active Directory External Authentication Plug-ins" on page 18-37.

4. Executing the oidspadi2.sh script creates two configuration entries: `cn=adwhencompare2,cn=plugin,cn=subconfigsubentry` and `cn=adwhenbind2,cn=plugin,cn=subconfigsubentry`. Use `ldapmodify` to disable or delete these entries.

5. Repeat the preceding steps for any additional domains, but be sure to use unique file names when you copy oidspada.pls and oidspadi.sh.

Enabling the Active Directory External Authentication Plug-ins

By default, the Active Directory external authentication plug-ins are enabled. However, you may need to enable them at some point.

To enable Active Directory external authentication plug-ins:

1. Create an LDIF file with the following entries:

```
dn: cn=adwhencompare,cn=plugin,cn=subconfigsubentry
changetype: modify
replace: orclpluginenable
orclpluginenable: 1

dn: cn=adwhenbind,cn=plugin,cn=subconfigsubentry
changetype: modify
replace: orclpluginenable
orclpluginenable: 1
```

2. Load the LDIF file with the `ldapmodify` command as follows:

```
ldapmodify -h host -p port  -D cn=orcladmin -w password -f fileName
```

See Also: The section about registering and managing plug-ins in *Oracle Internet Directory Administrator's Guide*

Testing the Active Directory External Authentication Plug-ins

To test the Active Directory external authentication plug-ins:

1. Use your browser to visit `http://`*`host of OracleAS Single Sign-On:port number of OracleAS Single Sign-On`*`/pls/orasso`.

2. Log in by using a pre-defined user in Microsoft Active Directory: `user identifier@domain`.

Configuring Windows Native Authentication

This section describes the system requirements and tasks for configuring Windows native authentication. It contains these topics:

- What are the System Requirements for Windows Native Authentication?

- Configuring Windows Native Authentication with a Single Microsoft Active Directory Domain
- Configuring Windows Native Authentication with Multiple Microsoft Active Directory Domains or Forests
- Implementing Fallback Authentication
- Understanding the Possible Login Scenarios

See Also: "Windows Native Authentication" on page 18-4

What are the System Requirements for Windows Native Authentication?

Windows native authentication is intended for intranet Web applications. Your intranet deployment must include the following:

- Windows 2000 server with Microsoft Active Directory
- Kerberos service account established for OracleAS Single Sign-On server
- Oracle Application Server 10*g* Release 2 (10.1.2) infrastructure installed

---

Note: Although the sample configurations in this section are for UNIX, Oracle Application Server can also be installed on Microsoft Windows.

- OracleAS Single Sign-On middle tier configured to use a Kerberos realm
- Synchronization of Active Directory with Oracle Internet Directory
- Oracle Internet Directory configured to use the Windows external authentication plug-in

Configuring Windows Native Authentication with a Single Microsoft Active Directory Domain

To set up Windows native authentication, configure Oracle Internet Directory, the OracleAS Single Sign-On server, and the user's browser by performing the following tasks in the order listed.

Task 1: Verify That Microsoft Active Directory Is Set Up and Working

To ensure that Microsoft Active Directory is properly configured and running, consult the Windows 2000/2003 server documentation.

Task 2: Install Oracle Internet Directory and OracleAS Single Sign-On

Install Oracle Internet Directory and OracleAS Single Sign-On. To determine which deployment configuration suits your installation, see the chapter about advanced configurations in *Oracle Application Server Single Sign-On Administrator's Guide*. For installation instructions, see the installation documentation for your operating system.

Task 3: Synchronize Oracle Internet Directory with Microsoft Active Directory

User entries in Oracle Internet Directory must be synchronized with user entries in Microsoft Active Directory.

Task 4: Configure the Active Directory External Authentication Plug-in for each Domain

This task is necessary to allow users to access Oracle Application Server Single Sign-On applications with browsers other than Internet Explorer 5.0 or later.

1. Install the Active Directory external authentication plug-in by following the instructions in "Configuring the Active Directory External Authentication Plug-in" on page 18-37.
2. Verify that the Active Directory external authentication plug-in is working by following the instructions in "Testing the Active Directory External Authentication Plug-ins" on page 18-39.

Task 5: Configure the OracleAS Single Sign-On Server

To configure the single sign-on server, complete the tasks described in the following topics.

- Set Up a Kerberos Service Account for the OracleAS Single Sign-On Server
- Run the OracleAS Single Sign-On Configuration Assistant on each Oracle Application Server Single Sign-On Host

Set Up a Kerberos Service Account for the OracleAS Single Sign-On Server Create a service account for the OracleAS Single Sign-On server in Active Directory, then create a keytab file for the server, and map the service principal (the server) to the account name. The keytab file stores the server's secret key. This file enables the server to authenticate to the KDC. The service principal is the entity, in this case, the single sign-on server, to which the KDC grants session tickets.

1. Synchronize system clocks. The OracleAS Single Sign-On middle tier and the Windows 2000 server must match. If you omit this step, then authentication fails because there is a difference in the system time.Be sure the time, the date, and the time zones are synchronized.
2. Check the port number of the Kerberos server on the Active Directory host. The port where the Kerberos server listens is selected from `/etc/services` by default. On Windows systems, the services file is found at *`system_drive`*`:\WINNT\system32\drivers\etc`. The service name is Kerberos. Typically the port is set to `88/udp` and `88/tcp` on the Windows 2000 server. When added correctly to the services file, the entries for these port numbers are:

```
kerberos5       88/udp          kdc             # Kerberos key server
kerberos5       88/tcp          kdc             # Kerberos key server
```

3. In the hosts file, located in the same directory as the services file, check the entry for the single sign-on middle tier. The fully qualified host name, which refers to the physical host name of the Oracle Application Server Single Sign-On server, must appear after the IP address and before the short name. The following is an example of a correct entry:

```
130.111.111.111 sso.MyCompany.com sso loghost
```

4. Perform the following tasks to create a user account and keytab file in Active Directory that will be used by the logical Oracle Application Server Single Sign-On host:
   a. Log in to the Active Directory Management tool on the Windows 2000 server; then choose Users, then New, then user.

Enter the name of the OracleAS Single Sign-On host, omitting the domain name. For example, if the host name is `sso.MyCompany.com`, then enter `sso`. This is the account name in Microsoft Active Directory.

Note the password that you assigned to the account. You will need it later. Do *not* select **User must change password at next logon**.

b. Create a keytab file for the OracleAS Single Sign-On server, and map the account name to the service principal name.You perform both tasks by running the following command on the Windows 2000 server:

```
C:> Ktpass -princ HTTP/sso.MyCompany.com@MyCompany.COM -pass password
-mapuser sso -out sso.keytab
```

The `-princ` argument is the service principal. Specify the value for this argument by using the format `HTTP/`*`single_sign-on_host_name`*`@`*`KERBEROS_REALM_NAME`*. Note that `HTTP` and the Kerberos realm must be uppercase.

Note that *`single_sign-on_host_name`* can be either the OracleAS Single Sign-On host itself or the name of a load balancer where multiple OracleAS Single Sign-On middle tiers are deployed. `MyCompany.COM` is a fictitious Kerberos realm in Microsoft Active Directory. The user container is located within this realm. The `-pass` argument is the account password that you obtained in Step 4. The `-mapuser` argument is the account name of the OracleAS Single Sign-On middle tier. You created this account in step 4. The `-out` argument is the output file that stores the service key.

Be sure to replace the example values given with values suitable for your installation. These values appear in boldface in the example.

Note:

- If the Ktpass is not found on your computer, then download the Windows resource kit to obtain the utility.
- The default encryption type for Microsoft Kerberos tickets is RC4-HMAC. Microsoft also supports DES-CBC and DES-CBC-MD5, two DES variants used in MIT-compliant implementations. Ktpass converts the key type of the KDC account from RC4_HMAC to DES.

---

5. For each Oracle Application Server Single Sign-On host, copy or FTP the keytab file, `sso.keytab`, created in step 5, to the OracleAS Single Sign-On middle tier, placing it in $*ORACLE_HOME*/j2ee/OC4J_SECURITY/config. If you use FTP, be sure to transfer the file in binary mode.

Be sure to give the Web server unique identifier (UID) on the OracleAS Single Sign-On middle tier read permission for the file.

Run the OracleAS Single Sign-On Configuration Assistant on each Oracle Application Server Single Sign-On Host Running the `ossoca.jar` tool at this point does the following:

- It configures the Oracle Application Server Single Sign-On server to use the Sun JAAS login module.
- It configures the server as a secured application.

To run the `ossoca.jar` tool on the OracleAS Single Sign-On middle tier:

1. Back up the following configuration files:
   - $*ORACLE_HOME*/sso/conf/policy.properties
   - $*ORACLE_HOME*/j2ee/OC4J_SECURITY/config/jazn.xml
   - $*ORACLE_HOME*/opmn/conf/opmn.xml
   - $*ORACLE_HOME*/j2ee/OC4J_SECURITY/config/jazn-data.xml
   - $*ORACLE_HOME*/j2ee/OC4J_SECURITY/applications/sso/web/WEB-INF/web.xml
   - $*ORACLE_HOME*/j2ee/OC4J_SECURITY/applications-deployments/sso/orion-application.xml
2. Run the ossoca.jar tool:
   - UNIX:

     ```
     $ORACLE_HOME/sso/bin/ssoca
     wna -mode sso
     -oh $ORACLE_HOME
     -ad_realm AD_REALM
     -kdc_host_port kerberos_server_host:port
     -verbose
     ```

   - Windows:

     ```
     %ORACLE_HOME%\jdk\bin\java -jar %ORACLE_HOME%\sso\lib\ossoca.jar
     wna -mode sso
     -oh %ORACLE_HOME%
     -ad_realm AD_REALM
     -kdc_host_port kerberos_server_host:port
     -verbose
     ```

   AD_REALM is the Kerberos realm in Microsoft Active Directory. This is the user container. Note from the syntax that this value must be entered in uppercase. The default port number for the KDC is usually 88. To confirm this, see step 2 in the section "Set Up a Kerberos Service Account for the OracleAS Single Sign-On Server" on page 18-41.
3. Step 2 shuts down the OracleAS Single Sign-On server. Restart it:

   ```
   $ORACLE_HOME/opmn/bin/opmnctl startall
   ```

Task 6: Configure Internet Explorer for Windows Native Authentication

Configure Internet Explorer to use Windows native authentication. How you do this depends on which version you have.

- Internet Explorer 5.0 and Later
- Internet Explorer 6.0 Only

Internet Explorer 5.0 and Later

To configure Internet Explorer and later, perform the following steps:

1. From the menu bar, select Tools, then, from the Tools menu, select Internet Options.
2. In the Internet Options dialog box, select the Security tab.
3. On the Security tab page, select Local Intranet, then select Sites.

4. In the Local intranet dialog box, select **Include all sites that bypass the proxy** server; then click **Advanced**.
5. In the advanced version of the Local intranet dialog box, enter the URL of the OracleAS Single Sign-On middle tier. For example:

   ```
   http://sso.mydomain.com
   ```

6. Click **OK** to exit the Local intranet dialog boxes.
7. In the Internet Options dialog box, select the **Security** tab; then choose **Local intranet**; then choose **Custom Level**.
8. In the Security Settings dialog box, scroll down to the User Authentication section and then select **Automatic logon only in Intranet zone**.
9. Click **OK** to exit the Security Settings dialog box.
10. From the menu bar, select Tools, then, from the Tools menu, select Internet Options.
11. In the Internet Options dialog box, select the Connections tab.
12. On the **Connections** tab page, choose **LAN Settings**.
13. Confirm that the correct address and port number for the proxy server are entered, then choose **Advanced**.
14. In the Proxy Settings dialog box, in the **Exceptions** section, enter the domain name for the OracleAS Single Sign-On server (`MyCompany.com` in the example).
15. Click **OK** to exit the Proxy Settings dialog box.

Internet Explorer 6.0 Only

If you are using Internet Explorer 6.0, perform steps 1 through 12 in "Internet Explorer 5.0 and Later"; then perform the following steps:

1. From the menu bar, select Tools, then, from the Tools menu, select Internet Options.
2. In the Internet Options dialog box, select the Advanced tab.
3. On the **Advanced** tab page, scroll down to the Security section.
4. Select **Enable Integrated Windows Authentication (requires restart)**.

Task 7: Reconfigure Local Accounts

After configuring Windows native authentication, you must reconfigure accounts for the Oracle Internet Directory administrator (`orcladmin`) and other local Windows users whose accounts are in Oracle Internet Directory. If you omit this task, then these users will not be able to log in.

Use the Oracle Directory Manager for Oracle Internet Directory to perform these steps:

1. Add the `orclADUser` class to the local user entry in Oracle Internet Directory.
2. Add the login ID of the local user to the `orclSAMAccountName` attribute in the user's entry. For example, the login ID of the orcladmin account is `orcladmin`.
3. Add the local user to the `exceptionEntry` property of the external authentication plug-in.

Configuring Windows Native Authentication with Multiple Microsoft Active Directory Domains or Forests

This section describes how to configure Windows native authentication with multiple Microsoft Active Directory domains or forests in the following types of deployments:

- Parent-child Microsoft Active Directory domains
- Microsoft Active Directory domains in the same forest with an established tree-root trust type
- Domains in different forests with an established forest trust type

> **Note:** Forest trust types are only supported in Windows Server 2003 and later versions of Windows operating systems.

To configure Windows native authentication with multiple Microsoft Active Directory domains or forests, perform the following tasks in the order listed:

Task 1: Verify that Trust is Established Between the Microsoft Active Directory Domains

Refer to your Microsoft Active Directory documentation for information on how to verify trust between multiple Microsoft Active Directory domains.

Task 2: Verify That Microsoft Active Directory Is Set Up and Working

To ensure that Microsoft Active Directory is properly configured and running, consult the Windows 2000/2003 server documentation.

Task 3: Install Oracle Internet Directory and OracleAS Single Sign-On

Install Oracle Internet Directory and OracleAS Single Sign-On. To determine which deployment configuration suits your installation, see the chapter about advanced configurations in *Oracle Application Server Single Sign-On Administrator's Guide*. For installation instructions, see the installation documentation for your operating system.

Task 4: Synchronize Oracle Internet Directory with each Microsoft Active Directory Domain

Create separate synchronization profiles for each Microsoft Active Directory by following the instructions described in "Configuring Synchronization Profiles" on page 18-17.

Task 5: Configure the Active Directory External Authentication Plug-in for each Domain

This task is necessary to allow users to access Oracle Application Server Single Sign-On applications with browsers other than Internet Explorer 5.0 or later.

1. Install the Active Directory external authentication plug-in for each domain by following the instructions in "Installing Active Directory External Authentication Plug-ins for Multiple Domains" on page 18-38.
2. Perform the following steps for each domain to verify that the Active Directory external authentication plug-in for each domain is working:
    1. Enter an `ldapbind` command to verify that a user entry was successfully imported from Active Directory into Oracle Internet Directory.

2. Enter an `ldapcompare` command to find whether `userPassword` attribute for the user entry exists in Oracle Internet Directory.

   See Also: The Oracle Internet Directory data management tools chapter in the *Oracle Identity Management User Reference* for information on the `ldapbind` and `ldapcompare` command-line utilities

Task 6: Enabling Windows Native Authentication with Oracle Application Server Single Sign-On through a Load Balancer or Reverse Proxy

Configure the Oracle Application Server Single Sign-On server to run behind a load balance or through reverse proxy by following the instructions in the advanced deployment options chapter of the *Oracle Application Server Single Sign-On Administrator's Guide*.

Task 7: Configure the OracleAS Single Sign-On Server

Configure each Oracle Application Server Single Sign-On server by following the instructions in "Task 5: Configure the OracleAS Single Sign-On Server" on page 18-41. Be sure to use the same Active Directory realm and corresponding key distribution center (KDC) when configuring each physical Oracle Application Server Single Sign-On server instance. Also, be sure to use the load balance or reverse proxy name as the logical Oracle Application Server Single Sign-On host name.

Note: With multiple Active Directory forests, the Oracle Application Server Single Sign-On server's logical host name must belong to one of the Active Directory domains. For example, assume you have two Active Directory forests and each forest contains a single domain. The domain in the first forest is named `engineering.mycompany.com` and the domain in the second forest is named `finance.mycompany.com`. The Oracle Application Server Single Sign-On server's logical host name must reside in either the `engineering.mycompany.com` or the finance.`mycompany.com` `domain`.

Task 8: Configure Internet Explorer for Windows Native Authentication

Configure the Oracle Application Server Single Sign-On server by following the instructions in "Task 6: Configure Internet Explorer for Windows Native Authentication" on page 18-43.

Implementing Fallback Authentication

Only browsers that are Internet Explorer 5.0 or later support SPNEGO-Kerberos authentication. OracleAS Single Sign-On provides fallback authentication support for unsupported browsers such as Netscape Communicator. Depending upon the type of browser and how it is configured, the user is presented with the OracleAS Single Sign-On login form or the HTTP basic authentication dialog box. In either case, the user must provide a user name and password. The user name consists of the Kerberos realm name and the user ID. The default way to enter the user name is shown in the following example.

`domain_name\user_id`

The following example, based on the example provided in "Set Up a Kerberos Service Account for the OracleAS Single Sign-On Server" on page 18-41, illustrates how to enter the user name.

```
MyCompany.COM\jdoe
```

Note that the user name and password are case sensitive. Additionally, password policies for Microsoft Active Directory do not apply. You can configure a different synchronization profile by using the Oracle directory integration and provisioning server. If you do, the login format just provided does not apply.

Fallback authentication is performed against Microsoft Active Directory, using an external authentication plug-in for Oracle Internet Directory.

Note:

- HTTP basic authentication does not support logout. To clear credentials from the browser cache, users must close all open browser windows. Alternatively, they can log out of the Windows computer.
- In cases where basic authentication is invoked, users must set their language preference manually in Internet Explorer. From the menu bar, select Tools; select Internet Options; select Languages; and then enter the desired language.

Understanding the Possible Login Scenarios

Users may encounter a number of different login behaviors within Internet Explorer depending upon which version they are using. Table 18–5 on page 18-47 shows under what circumstances automatic sign-on and fallback authentication are invoked.

Table 18–5 Single Sign-On Login Options in Internet Explorer

Browser Version	Desktop Platform	Desktop Authentication Type	Integrated Authentication in Internet Explorer Browser	OracleAS Single Sign-On Login Type
5.0.1 or later	Windows 2000/XP	Kerberos V5	On	Automatic sign-on
5.0.1 or later but earlier than 6.0	Windows 2000/XP	Kerberos V5	Off	Single sign-on
6.0 or later	Windows 2000/XP	Kerberos V5 or NTLM	Off	HTTP basic authentication
5.0.1 or later but earlier than 6.0	Windows NT/2000/XP	NTLM	On or off	Single sign-on
6.0 or later	NT/2000/XP	NTLM	On	Single sign-on
5.0.1 or later	Windows 95, ME, Windows NT 4.0	Not applicable	Not applicable	Single sign-on
Earlier than 5.0.1	N/A	Not applicable	Not applicable	Single sign-on
All other browsers	All other platforms	Not applicable	Not applicable	Single sign-on

Configuring Synchronization of Oracle Internet Directory Foreign Security Principal References with Microsoft Active Directory

This section explains how to synchronize Oracle Internet Directory foreign security principal references with Active Directory.

Although Microsoft Active Directory stores information for group members in a trusted domain as foreign security principal references, Oracle Internet Directory stores the DNs of these members as they appear in Oracle Internet Directory. This results in a mismatch between an entry and its value as a member of a group. The relationship between a user and a group cannot be directly established in Oracle Internet Directory.

To establish the relationship between users and groups, the member DNs that refer to the foreign security principals must be replaced by the DNs of the entries during the synchronization of such groups. This is called resolving foreign key references.

Note: Synchronization of foreign security principal references is supported only on Windows 2003.

Example 18–5 How Foreign Key References Are Resolved

The example in this section illustrates how foreign key references are resolved.

Assume that there are three domains: A, B and C.

Domain A has a one-way non-transitive trust to Domain B. It can have foreign security principal references for users and groups from Domain B.

Domain A has a one-way non-transitive trust to Domain C. It can have foreign security principal references for users and groups from Domain C.

Domain B has a one-way non-transitive trust to Domain C. It can have foreign security principal references for users and groups from Domain C.

In this example, the one-way non-transitive trusts are from Domain A to Domain B, from Domain A to Domain C, and from Domain B to Domain C.

Tasks to Resolve Foreign Key References

This section explains the steps for resolving foreign key references.

Task 1: Update Agent Configuration Information For each profile that can have foreign security principal references, perform the following steps. The sample configuration files referred further are available in $*ORACLE_HOME*/ldap/odi/samples directory.

1. Copy the `activeimp.cfg.fsp` file. The following is an example of the `activeimp.cfg.fsp` file:

```
[INTERFACEDETAILS]
   Package: gsi
   Reader: ActiveReader
[TRUSTEDPROFILES]
   prof1 : <Name of the profile1>
   prof2 : <Name of the profile2>
[FSPMAXSIZE]
   val=10000
```

The preceding example assumes you are using the DirSync change tracking approach. If you are using the USN-Changed approach for tracking changes, assign a value of `ActiveChgReader` to the Reader parameter.

2. In the `activeimp.cfg.fsp` file, under the `[TRUSTEDPROFILES]` tag, specify the profile names of the other domains that have foreign security principal references in this domain.

   Referring to Example 18–5 on page 18-48, agent configuration information for Domain A contains the following:

```
[INTERFACEDETAILS]
   Package: gsi
   Reader: ActiveReader
[TRUSTEDPROFILES]
   prof1: profile_name_for_domain_B
   prof2: profile_name_for_domain_C
```

   Agent configuration information for domain B contains the following:

```
[INTERFACEDETAILS]
   Package: gsi
   Reader: ActiveReader
[TRUSTEDPROFILES]
   prof1: profile_name_for_domain_C
```

   Agent configuration information for domain C has no changes because domain C has no foreign key references.

3. Under the `[FSPMAXSIZE]` tag, specify the foreign security principal cache size. This can be the average number of foreign security principals you can have. A sample value of 1000 is specified in the `activeimp.cfg.fsp` file.

4. Load the new agent configuration information file by using the Directory Integration and Provisioning Assistant as follows:

```
$ORACLE_HOME/bin/dipassistant modifyprofile
-profile profile_name_for_domain_A_or_B
-host host_name
-port port_name
-dn bind_DN
-passwd password_of_bind_DN
odip.profile.configfile=activeimp.cfg.fsp
```

5. Repeat this task for every profile of interest.

Task 2: Modify the Input Data Before Bootstrapping to Resolve the Foreign Security Principal References To do this, perform the following steps:

1. Get the LDIF dump from the Active Directory with appropriate filtering so that the resultant LDIF file contains only the required objects, for example users and groups.

   Note: The command to dump entries from Microsoft Active Directory to Oracle Internet Directory is `ldifde`. This command can be run only from a Microsoft Windows environment.

2. Resolve the foreign security principal references by entering the following command:

```
$ORACLE_HOME/ldap/odi/admin/fsptodn
host=oid_host
port=oid_port
dn= OID_privileged_DN (that is, superuser or dipadmin user)
pwd=OID_password
profile=profile_name_for_domain_A_or_B
infile=input_filenameo_of_the_LDIF_dump_from_Active_Directory
outfile=output_filename
[sslauth=0|1]
```

By default, `host` is set to *`local_host`*, `port` is set to `389`, and `sslauth` is set to `0`.

Note: You can verify the successful execution of the command by verifying that the output file contains no references to `cn=foreignsecurityprincipals` in the member attribute. This command performs no attribute-level mapping other than resolving foreign security principal references.

---

3. Use the `-bootstrap` option of the Directory Integration and Provisioning Assistant to bootstrap the data from Microsoft Active Directory to Oracle Internet Directory.

   See Also: "Bootstrapping Data Between Directories" on page 18-52

Task 3: Update the Mapping Rules to Resolve the Foreign Security Principals During Synchronization After bootstrapping, modifications to groups must be reflected in Oracle Internet Directory with the correct group membership values. The `fsptodn` mapping rule enables you to do this when you synchronize. Modify this mapping rule in every profile that needs foreign security principal resolution. Referring to Example 18–5 on page 18-48, the mapping rules must be modified for Domains A and B.

If you do not have DN mapping, then change your mapping rule for the `member` attribute to the following:

```
member: : :group:uniquemember: :groupofUniqueNames: fsptodn(member)
```

If you have DN mapping, then change the mapping rules as follows:

1. Add the DN mapping rules corresponding to each of the trusted domains. This is used to resolve the correct domain mapping. Referring to Example 18–5 on page 18-48, the `domainrules` in the mapping file for Domain A should have content similar to the following:

   ```
   DOMAINRULES
   <Src Domain A >:<Dst domain A1 in OID>
   <Src Domain B >:< Dst domain B1 in OID>
   <Src Domain C>:<Dst domain C1 in OID>
   ```

2. Change your mapping rule for the `member` attribute to:

   ```
   member:::group:uniquemember::groupofUniqueNames:dnconvert(fsptodn(member))
   ```

3. Upload the mapping file for the different profiles using Directory Integration and Provisioning Assistant.

Managing Integration with Microsoft Active Directory

This section describes what to do immediately after configuration and ongoing administration tasks. It contains these topics:

- Tasks After Configuring with Microsoft Active Directory
- Typical Management of Integration with Microsoft Active Directory

Tasks After Configuring with Microsoft Active Directory

Once configuration is complete, do the following:

1. Migrate data from one directory to the other as needed. This is described in "Bootstrapping Data Between Directories" on page 18-52.
2. Enable the integration profile. You can do this by using either the Oracle Directory Integration and Provisioning Server Administration tool or the command-line version of the Directory Integration and Provisioning Assistant.

   To enable the integration profile by using the Oracle Directory Integration and Provisioning Server Administration tool, perform the following:

   a. Launch the Oracle Directory Integration and Provisioning Server Administration by entering the following:

      ```
      $ORACLE_HOME/bin/dipassistant -gui
      ```

   b. In the navigator pane, expand ***directory_integration_and_provisioning_server***, then expand **Integration Profile Configuration**.

   c. In the navigator pane, select the configuration set. A list of the available profiles appears in the right pane.

   d. In the right pane, select the profile, then choose Edit. The General tab page window appears.

   e. In the General tab page, in the Profile Status field, select ENABLE.

   f. Choose OK.

   To enable the synchronization profile by using the command-line version of the Directory Integration and Provisioning Assistant, enter the following command:

   ```
   $ORACLE_HOME/bin/dipassistant modifyprofile
   [-h host name] [-p port_number] [-D bind_DN] [-w password]
   -profile profile_name_in_OID odip.profile.status=ENABLE
   [-configset configset_number]
   ```

3. Start the Oracle directory integration and provisioning server using the configuration set that corresponds to that of the profile. See "Starting, Stopping, and Restarting the Oracle Directory Integration and Provisioning Server" on page 4-8.

Typical Management of Integration with Microsoft Active Directory

Management tasks typically include:

- Managing synchronization profiles and mapping rules:
  - Creating new profiles. You create new profiles if you need to synchronize with an additional domain controller in a multiple domain Active Directory environment.

You can create new profiles by using existing profiles as templates.To do this, use the `createlike` command of the Directory Integration and Provisioning Assistant.

  - Changing configurations (attributes) in the profile
  - Disabling profiles to allow maintenance and then reenabling them. Disabling profiles stops synchronization related to that profile.
- Managing mapping rules:
  - Creating new rules when additional attributes need to be synchronized
  - Changing existing rules when the way attributes are synchronized needs to change
  - Deleting or commenting out rules not required when a particular attribute is not required to be synchronized
- Managing access control
- Starting and stopping the Oracle directory server and the Oracle directory integration and provisioning server

This section contains these topics:

- Bootstrapping Data Between Directories
- Managing the Active Directory External Authentication Plug-in
- Switching to a Different Microsoft Active Directory Domain Controller in the Same Domain

See Also:

- "Configuring Synchronization Profiles" on page 18-17 for instructions on managing profiles, mapping rules, and access control
- *Oracle Identity Management User Reference* for instructions on starting and stopping servers

Bootstrapping Data Between Directories

Bootstrapping is sometimes called data migration. To bootstrap data, do the following once the Active Directory Connector and plug-in configurations are complete:

1. Identify the data you want to migrate. You can choose to migrate all data in the directory or only a subset of data.
2. Make sure the synchronization is not enabled yet.
3. Bootstrap from one directory to another by using the Directory Integration and Provisioning Assistant with the `-bootstrap` option. Bootstrapping is described in Chapter 8, "Bootstrapping of a Directory in Oracle Directory Integration and Provisioning".

   Once bootstrapping is accomplished, the profile status attributes are appropriately updated in the synchronization profile by the Directory Integration and Provisioning Assistant.
4. If you used LDIF file-based bootstrapping, then initialize the `lastchangekey` value with the Directory Integration and Provisioning Assistant as follows:

   ```
   $ORACLE_HOME/bin/dipassistant modifyprofile -updlcn
   ```

This `lastchangekey` attribute should be set to the value of the last change number in the source directory before you started the bootstrap.

In order to update the last change number, the value assigned to the `odip.profile.condirurl` property in the import synchronization profile must be for a non-SSL connection. If you have already configured the import synchronization profile for SSL, then before attempting to update the last change number, you must temporarily change the value assigned to the `odip.profile.condirurl` property so it points to a non-SSL port.

5. If two-way synchronization is required, then enable the export profile and make sure the change logging option is enabled for the Oracle directory server. Change logging is controlled by the `-l` option while starting Oracle Internet Directory. By default, it is set to `TRUE`, meaning that change logging is enabled. If it is set to `FALSE`, then use the OID Control Utility to shut down the Oracle directory server, and then to start the server again with the change log enabled.

Managing the Active Directory External Authentication Plug-in

This section explains how to delete, disable, and re-enable the Active Directory external authentication plug-in.

Deleting the Active Directory External Authentication Plug-in To delete the Active Directory external authentication plug-in, enter the following commands:

```
ldapdelete -h host -p port -D cn=orcladmin -w password
"cn=adwhencompare,cn=plugin,cn=subconfigsubentry"

ldapdelete -h host -p port -D cn=orcladmin -w password
"cn=adwhenbind,cn=plugin,cn=subconfigsubentry"
```

Disabling the Active Directory External Authentication Plug-in To disable the Microsoft Active Directory external authentication plug-in:

1. Create an LDIF file with the following entries:

```
dn: cn=adwhencompare,cn=plugin,cn=subconfigsubentry
changetype: modify
replace: orclpluginenable
orclpluginenable: 0

dn: cn=adwhenbind,cn=plugin,cn=subconfigsubentry
changetype: modify
replace: orclpluginenable
orclpluginenable: 0
```

2. Load the LDIF file with the `ldapmodify` command, as follows:

```
ldapmodify -h host -p port -D cn=orcladmin -w password -f fileName
```

Reenabling the Active Directory External Authentication Plug-in To re-enable the Active Directory external authentication plug-in, use these two commands:

1. Create an LDIF file with the following entries:

```
dn: cn=adwhencompare,cn=plugin,cn=subconfigsubentry
changetype: modify
replace: orclpluginenable
orclpluginenable: 1

dn: cn=adwhenbind,cn=plugin,cn=subconfigsubentry
changetype: modify
```

```
replace: orclpluginenable
orclpluginenable: 1
```

2. Load the LDIF file with the `ldapmodify` command, as follows:

```
ldapmodify -h host -p port -D cn=orcladmin -w password -f fileName
```

See Also:

- "Configuring the Active Directory External Authentication Plug-in" on page 18-37
- "Enabling the Active Directory External Authentication Plug-ins" on page 18-39

Switching to a Different Microsoft Active Directory Domain Controller in the Same Domain

This section explains how to change the Microsoft Active Directory domain controller to which changes are exported. There are two methods, one for the USN-Changed approach and the other for the DirSync approach.

How to Change the Active Directory Domain Controller by Using the USN-Changed Approach

If you are using the USN-Changed approach, then perform the following:

1. Stop the current running profile. Modify the Microsoft Active Directory host connection information, that is, host, port, user, password, to point to the new host. Usually, the host name is the only item that you need to update.
2. Obtain the current value of the `highestCommittedUSN` by searching the new domain controller's root DSE for the current highest `uSNChanged` value (attribute value of the `highestCommittedUSN` attribute of the root DSE):

```
ldapsearch -h host -p port -b "" -s base -D user
DN -w password "objectclass=*" highestCommittedUSN
```

3. Use Oracle Directory Integration and Provisioning to run a full synchronization from Microsoft Active Directory.
   a. Run ldifde, the command to dump entries from Microsoft Active Directory to Oracle Internet Directory, using the intended ldapsearch scope and search filter. Normally, the search filter should be the same as that specified in the running profile. For example, the following search filter is set in the sample properties file in Release 10.1.2: Note that ldifde can be run only from a Microsoft Windows environment.

```
searchfilter=(&(|(objectclass=user)(objectclass=organizationalunit))(!(obje
ctclass=group)))
```

      Essentially, run ldifde with a search scope and search filter that retrieve all Oracle Internet Directory objects (entries) that were configured to be synchronized with Microsoft Active Directory by the running profile.

   b. Run Oracle Directory Integration and Provisioning to upload the LDIF file generated in Step a using the same profile.
4. After the full synchronization is completed, update the `lastchangenumber` attribute with the `highestCommittedUSN` value obtained in Step 2.
5. Resume the normal synchronization, that is, incremental synchronization from Microsoft Active Directory using `uSNChanged` attribute.

How to Change the Active Directory Domain Controller by Using the DirSync Approach

If you are using the DirSync approach, perform the following:

1. Stop the current profile that is running.
2. Use the Directory Integration and Provisioning Assistant `createlike` option to create a new profile exactly the same as the profile already being used. In the newly created profile, modify the Microsoft Active Directory host connection information, that is, host, port, user, password, to point to the new host. Usually, the host name is the only item you need to update.
3. Resume normal synchronization with the modified profile. Note all the domain controllers must be in the same Active Directory domain.

19

Integration with the Microsoft Windows NT 4.0 Environment

This chapter explains how to integrate Oracle Identity Management with Microsoft Windows NT 4.0. It contains these topics:

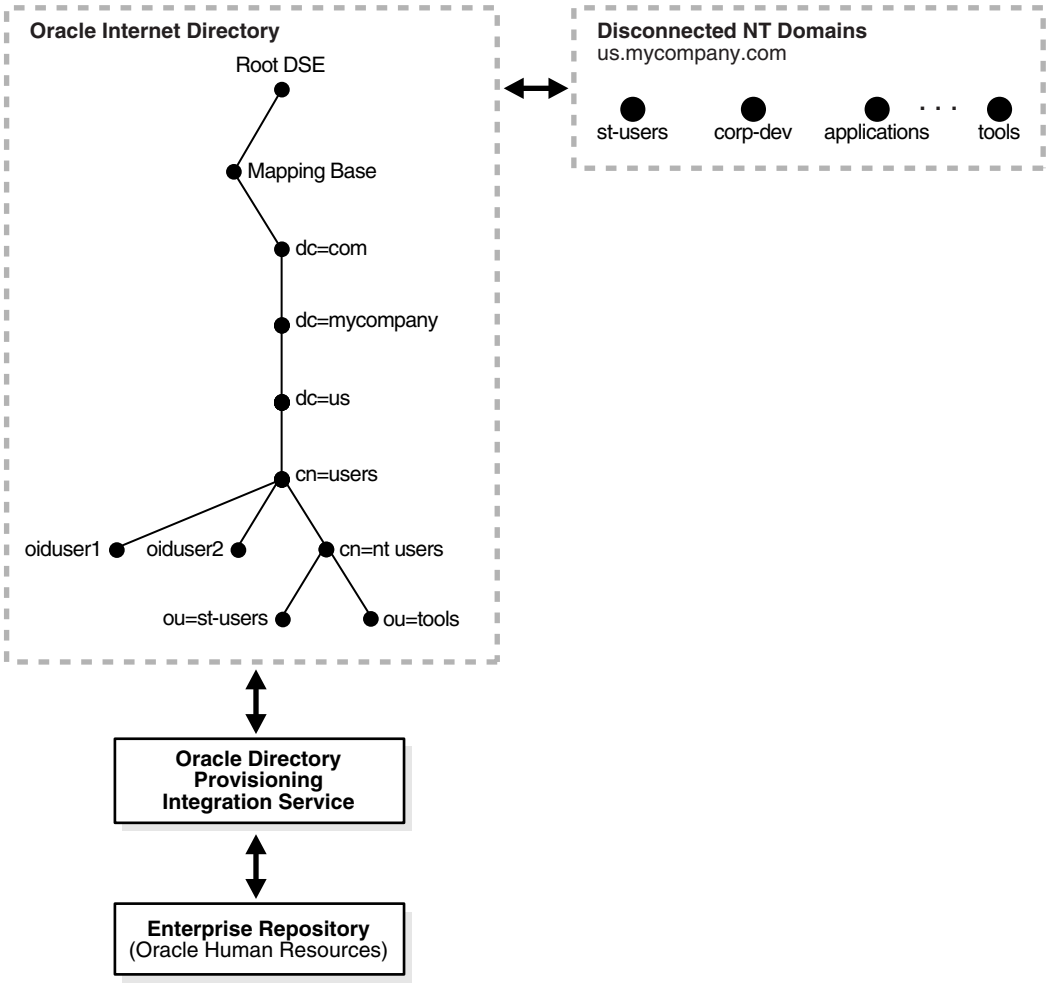
- Overview of Integration with Microsoft Windows NT 4.0
- Installing and Configuring Windows NT External Authentication and Auto-Provisioning Plug-ins

> **See Also:** Chapter 18, "Integration with the Microsoft Active Directory Environment"

Overview of Integration with Microsoft Windows NT 4.0

Microsoft Windows NT domain users can be integrated with Oracle Identity Management. Microsoft Windows NT groups are not synchronized to Oracle Internet Directory, nor is information about the members of those groups. In this case, each of the Microsoft Windows NT domains can be mapped to a domain object or an organization unit object in Oracle Internet Directory. Typical mapping of Microsoft Windows NT domains to domain containers in the Oracle Internet Directory directory information tree is shown in Figure 19–1.

Figure 19–1 Integration of Oracle Internet Directory DIT with Microsoft Windows NT Domains



Microsoft Windows NT domains are integrated with Oracle Internet Directory so that a minimal user footprint is automatically created in Oracle Internet Directory.

If a user entry exists in Microsoft Windows NT but not in Oracle Internet Directory, then, when that user tries to log in to use the Oracle Application Server components, the auto-registration plug-in creates a shadow entry with minimal footprint information in Oracle Internet Directory. This entry remains in Oracle Internet Directory for the next time the same user tries to log in.

External authentication, with Microsoft Windows NT acting as the external repository, is supported by the use of plug-ins. Ongoing synchronization with the Microsoft Windows NT environment is not supported.

Installing and Configuring Windows NT External Authentication and Auto-Provisioning Plug-ins

This section contains these topics:

- Installing and Enabling the Windows NT External Authentication and Provisioning Plug-ins
- Managing the Windows NT External Authentication and Provisioning Plug-ins

Installing and Enabling the Windows NT External Authentication and Provisioning Plug-ins

The SQL script oidspnti.sql installs the plug-ins that enable Oracle Internet Directory for external authentication against the Microsoft Windows primary domain controller and auto-provisioning.

Note: To run shell script tools on the Windows operating system, you need one of the following UNIX emulation utilities:

- Cygwin 1.3.2.2-1 or later. Visit: `http://sources.redhat.com`
- MKS Toolkit 6.1. Visit: `http://www.datafocus.com/`

To install the script:

1. Verify that the Oracle Internet Directory server is running.
2. Run the script by entering the following commands:

   ```
   cd $ORACLE_HOME/ldap/admin
   sh oidspnti.sh
   ```

3. Enter the Oracle Internet Directory host name and port number. The default port number is 389.
4. Enter the password of the Oracle administrator (`orcladmin`), the directory super user.
5. Enter the distinguished name of the container to which the plug-in needs to be applied. Every entry in this container is then authenticated against the Microsoft Windows NT domain. Note that this need not necessarily be the user search base supplied in the Oracle Internet Directory Self-Service Console. All the users under this search base are authenticated externally to the Microsoft Windows NT domain. If more than one value is specified, then use semi-colons (`;`) to separate them.
6. Enter the plug-in request group DN. For security reasons, the plug-in can be invoked only by users belonging to this group. For example, suppose that the Oracle Application Server Single Sign-On administrators are in the group `cn=OracleUserSecurityAdmins,cn=Groups,cn=OracleContext`. If you enter this value for the plug-in request group DN, then only the requests coming from Oracle Application Server Single Sign-On administrators can trigger the external authentication plug-in. You can enter multiple DN values. Use a semicolon (;) to separate them. This value is not required, but, for security purposes, should be specified.
7. Choose Auto Registration. The default is Yes. Upon registration, each entry is assigned the object class `orclNTUser`.

At the completion of these steps, the plug-ins are installed.

Managing the Windows NT External Authentication and Provisioning Plug-ins

This section tells you how to:

- Enable and disable the plug-ins
- Enable and disable auto-provisioning

- Remove the plug-ins
- Debug the Windows NT external authentication plug-in

Enabling the Windows NT External Authentication Plug-in

To enable external authentication, enter these two commands:

```
ldapmodify -h host -p port -D cn=orcladmin -w password <<EOF
dn: cn=ntwhencompare,cn=plugin,cn=subconfigsubentry
changetype: modify
replace: orclpluginenable
orclpluginenable: 1
EOF

ldapmodify -h host -p port -D cn=orcladmin -w password <<EOF
dn: cn=ntwhenbind,cn=plugin,cn=subconfigsubentry
changetype: modify
replace: orclpluginenable
orclpluginenable: 1
EOF
```

Disabling the Windows NT External Authentication Plug-in

To disable the external authentication plug-ins, set the value of the attribute orclpluginenable to 0 in each of the preceding command.

Enabling Auto-Provisioning

To enable auto provisioning, enter the following command:

```
ldapmodify -h host -p port -D cn=orcladmin -w password <<EOF
dn: cn=ntpostsearch,cn=plugin,cn=subconfigsubentry
changetype: modify
replace: orclpluginenable
orclpluginenable: 1
EOF
```

Disabling Auto-Provisioning

To disable auto provisioning, use the previous command, but set the value of the attribute `orclpluginenable` to `0`.

Removing Windows NT External Authentication and Auto-Provisioning Plug-ins

To remove external authentication and auto-provisioning, delete the two plug-in entries from Oracle Internet Directory:

```
ldapdelete -h host -p port D cn=orcladmin -w password
"cn=ntwhencompare,cn=plugin,cn=subconfigsubentry"

ldapdelete -h host -p port D cn=orcladmin -w password
"cn=ntwhenbind,cn=plugin,cn=subconfigsubentry"

ldapdelete -h host -p port D cn=orcladmin -w password
"cn=ntpostsearch,cn=plugin,cn=subconfigsubentry"
```

Debugging the Windows NT External Authentication Plug-in

If you are experiencing unknown errors, then you can enable the plug-in debugging. To do this, enter:

```
sqlplus ods/odspassword @$ORACLE_HOME/ldap/admin/oidspdon.pls
```

To check the plug-in debugging log:

```
sqlplus ods/ods
select * from plg_debug_log order by id;
```

To delete the plug-in debugging log:

```
sqlplus ods/ods
truncate table plg_debug_log
```

To disable plug-in debugging:

```
sqlplus ods/ods @$ORACLE_HOME/ldap/admin/oidspdof.pls
```

Note: If you need to change the Windows NT external authentication plug-in setup—that is, the information you entered in the installation steps—then rerun the installation script. Before you rerun the script, remove the Windows NT external authentication plug-ins by following the preceding instructions.

20

Integration with SunONE (iPlanet) Directory Server

This chapter explains how to integrate the Oracle Identity Management infrastructure with SunONE Directory Server (Netscape Directory Server and iPlanet Directory Server) by using the SunONE connector in Oracle Directory Integration and Provisioning.

> **Note:** This chapter assumes that you have read Chapter 17, "Considerations for Integrating with Third-Party Directories" and made the necessary deployment decisions and basic configurations.

This chapter contains these topics:

- About the SunONE Connector
- Configuring the SunONE Connector
- The Synchronization Process
- Supported Configurations for Integrating with SunONE Directory Server

> **See Also:** "Troubleshooting Integration with the SunONE Connector" on page C-26

About the SunONE Connector

The SunONE connector includes a synchronization component that is driven by the Oracle directory integration and provisioning server. This component maintains consistency between the directories by:

- Importing data and incremental changes from an SunONE Directory Server into Oracle Internet Directory
- Exporting data and incremental changes from Oracle Internet Directory into an SunONE Directory Server

The SunONE Directory Server and Oracle Internet Directory support similar hashing techniques for storing passwords. If the mapping rules are configured appropriately, then the SunONE connector can synchronize passwords, the same as any other attribute. In this case, passwords are in the hashed format. However, if you store passwords only in the SunONE Directory Server, use the SunONE Directory Server external authentication plug-in discussed in this chapter.

Note: Oracle Internet Directory 10*g* Release 2 (10.1.2) can synchronize with Netscape Directory Server Release 4.13 and SunONE (iPlanet) Directory Server Releases 5.0, 5.1, and 5.2

See Also:

- "Choose Where to Store Passwords" on page 17-5
- The chapter on directory storage of password verifiers in *Oracle Internet Directory Administrator's Guide* for a list of hashing algorithms supported by Oracle Internet Directory

SunONE Directory Server Integration Concepts

This section contains these topics:

- Synchronization Between Oracle Internet Directory and SunONE Directory Server
- Synchronization of Deletions from SunONE Directory Server to Oracle Internet Directory
- The SunONE Directory Server External Authentication Plug-in

Synchronization Between Oracle Internet Directory and SunONE Directory Server

Synchronization with SunONE Directory Server is based on reading incremental changes from the source directory to the destination directory. If changes are to be made in both directories, then both directories need to have change logging enabled.

See Also:

- The Oracle Internet Directory server administration tools chapter of the *Oracle Identity Management User Reference* for instructions on how to start an Oracle directory server with change logging enabled.
- SunONE Directory Server documentation for instructions on how to configure change logging. If you plan to synchronize with SunONE (iPlanet) Directory Server Release versions 5.0 or later, the retro changelog plug-in must be enabled.

Synchronization of Deletions from SunONE Directory Server to Oracle Internet Directory

If you want to synchronize deletions, and the mapping rules have mandatory attributes, then be sure that the tombstone is configured correctly.

To verify that the tombstone is configured in SunONE Directory Server, execute the following command:

```
$ORACLE_HOME/bin/ldapsearch -h connected_directory_host
-p connected_directory_port -D  connected_directory_account
-w connected_directory_password -b source_domain
-s sub "objectclass=nstombstone"
```

This returns information on all deleted entries.

See Also: SunONE documentation for details about configuring tombstones

Note: Tombstones are automatically configured on the SunONE Directory Server if replication is enabled.

The SunONE Directory Server External Authentication Plug-in

Oracle components are clients of Oracle Internet Directory. However, in an integrated environment, you have the option of storing security credentials for those components in an external repository —in this case, SunONE Directory Server—rather than in Oracle Internet Directory. When security credentials are stored in an external repository, user authentication to an Oracle component happens in the external repository and not in Oracle Internet Directory.

To communicate with the external repository, the Oracle component relies on the Oracle directory server. The Oracle directory server, in turn, uses a plug-in that can access the external repository. The entire authentication process is transparent to the Oracle components, which perceive all the LDAP requests as being handled by the Oracle directory server.

Types of External Authentication

To verify a user's security credentials, an Oracle component can, by way of the Oracle directory server, send to the external repository a simple bind with a request for one of the following:

- Non-SSL ldapbind
- SSL ldapbind
- ldapcompare

How Authentication to an External Repository Works

When an Oracle directory server has the plug-in configured and enabled, the following process occurs to authenticate a user to an Oracle component.

1. The user seeks access to an Oracle component.
2. The Oracle component, which is a client of Oracle Internet Directory, receives the authentication request, and passes to the Oracle directory server either an ldapbind or ldapcompare request.
3. The Oracle directory server passes the control to the plug-in.
4. The plug-in issues the request to the external repository.
5. The plug-in obtains the results of that request and passes the results back to the Oracle directory server.
6. The Oracle directory server passes the results back to client application, which then grants or denies access to the user.

Configuring the SunONE Connector

This section explains the tasks to configure the SunONE connector. It contains these topics:

- Task 1: Configure the Synchronization Profiles for the SunONE Connector

- Task 2: Configure Access Control Lists
- Task 3: Prepare Both Directories for Synchronization
- Task 4: (Optional) Configure the SunONE Directory Server External Authentication Plug-in
- Task 5: Start the Synchronization

Task 1: Configure the Synchronization Profiles for the SunONE Connector

The following two default Integration profiles for synchronization with the SunONE Directory Server are created in the Oracle directory server as a part of the installation process:

- `iPlanetImport`—for importing entries and changes from the SunONE Directory Server by using the directory synchronization approach
- `iPlanetExport`—for exporting changes from Oracle Internet Directory to SunONE Directory Server

> **See Also:** Chapter 6, "Configuration of Directory Synchronization Profiles"

Customizing the Default Integration Profiles

Although you can enable synchronization with the SunONE Directory Server by customizing the default `iPlanetImport` and `iPlanetExport` integration profiles, the recommended approach is to create new profiles based on the default integration profiles. You can use either the Directory Integration and Provisioning Assistant's `createprofilelike` command or the Oracle Directory Integration and Provisioning Server Administration tool to create new profiles based on existing profiles.

To use the Directory Integration and Provisioning Assistant's `createprofilelike` command to create new profiles based on the existing default integration profiles, use the following syntax:

```
dipassistant createprofilelike [-h hostName] [-p port] [-D bindDn]
[-w password] -profile origProfName -newprofile newProfName
```

Use the preceding command to make copies of both the `iPlanetImport` and `iPlanetExport` integration profiles.

To use the Oracle Directory Integration and Provisioning Server Administration to create new profiles based on the existing default integration profiles:

1. Launch the Oracle Directory Integration and Provisioning Server Administration tool by entering:

   ```
   $ORACLE_HOME/bin/dipassistant -gui
   ```

2. In the navigator pane, expand ***directory_integration_and_provisioning_server***, then expand **Integration Profile Configuration**.
3. Select the configuration set, and, in the right pane, choose **Create**. The Integration Profiles window appears.

   This window is described in "Integration Profiles" on page A-6.

4. In the Integration Profile window, select the **IplanetImport** or **IplanetExport** profile, and then choose **Create Like**. The General tab of the Integration Profile window appears.

   This tab is described in "General" on page A-6.

5. Enter a name for the new profile and make any additional changes in the General tab or other tabs in the Integration Profile window to finish customizing the profile.

6. Choose **OK**.

Configuring the Connection Details for the SunONE Directory Server

You must update the SunONE Directory server connection details in the synchronization profiles as follows:

1. Create a user account in the SunONE Directory server with administrative privileges. Oracle Directory Integration and Provisioning will use this account to connect to SunONE Directory server. You must grant sufficient privileges to perform both import and export operations.

   - **For Import Operations from SunONE Directory Server**: Grant the user account the following permissions:
     - Permissions to read the change log entry
     - Permissions to read the tombstone
     - Permissions to read the entries under the container to be synchronized
   - **For Export Operations to SunONE Directory Server**: Grant the user account write permission to the subtree root that is the parent of all the containers to which the Oracle directory integration and provisioning server will export users.

2. Update the connection details in the `odip.profile.condirurl`, `odip.profile.condiraccount`, and `odip.profile.condirpassword` properties of the synchronization profiles. You can use either the Directory Integration and Provisioning Assistant or the Oracle Directory Integration and Provisioning Server Administration tool.

See Also:

- Chapter 3, "Oracle Directory Integration and Provisioning Administration Tools"
- The Oracle Directory Integration and Provisioning tools chapter in the *Oracle Identity Management User Reference*

Configuring the Default Integration Profile Through the Script iplconfig.sh

Use this method when:

- The SunONE Directory Server has no custom schema changes to the objects to be synchronized—that is, the user and group object attributes and object classes are the default ones
- No custom schema elements have been added to the user or group object attributes and object classes

At the end of synchronization, user and group objects synchronized from the SunONE Directory Server are visible to Oracle components integrated with the Oracle Application Server infrastructure.

The script `iplconfig.sh` resides in $*ORACLE_HOME*/ldap/odi/admin. It prompts you for the following:

- Oracle Internet Directory super user DN and password
- SunONE Directory Server URL (*`host:port`*)
- SunONE Directory Server user account and password to be used by the SunONE connector
- SunONE Directory Server domain to be synchronized

Once you have entered the parameter values, `iplconfig.sh` invokes the Directory Integration and Provisioning Assistant to set up the SunONE Directory Server connection information and mapping rules information in the default SunONE Directory Server integration profiles.

Configuring Password Synchronization

The default mapping rules are not appropriate for password synchronization between the SunONE Directory Server and Oracle Internet Directory.

If Oracle Internet Directory and the SunONE Directory Server use the same password hashing technique, then insert the following mapping rule to the mapping file and upload the mapping file to the profile.

```
Userpassword: : :person:userpassword: :person
```

If the two directories do not use the same hashing technique, then the same mapping rule works when the Oracle directory integration and provisioning server and the directory integration profile are configured in SSL mode 2—that is, server-only authentication.

Configuring the Integration Profiles for Two-Way Synchronization

If you have two-way synchronization enabled, then you need to avoid having the same changes synchronized back and forth between the directories by setting the filter attributes for the connected directory and for Oracle Internet Directory. You can use either the Oracle Directory Integration and Provisioning Server Administration tool or the Directory Integration and Provisioning Assistant to perform this task.

In the import profile, set the connected directory filter as follows:

```
modifiersname != <DN of the user account with which changes are made by the export
profile in SunONE>
```

In the export profile, set the Oracle Internet Directory filter as follows:

```
modifiersname != orclodipagentname=<import profile name>,cn=subscriber
profile,cn=changelog subscriber,cn=oracle internet directory
```

Configuring Mapping Rules

The default profiles have the default mapping rules for mapping the user and group attributes and object classes in SunONE Directory Server to those on Oracle Internet Directory. These mapping rules assume that no user- and group-specific schema changes have been made to either directory after installation. If there are such changes, then they must be appropriately reflected in the mapping files.

To verify and modify the mapping rules, do the following:

1. Decide which domains, or containers, you want to synchronize. In the case of SunONE Directory Server, the container to be specified for synchronization can be any naming context in the directory.
2. Decide on the objects—that is, the types of entries—to be synchronized. In an identity management environment these are typically user and group entries.
3. Identify the attributes and how you want to map them between the directories during synchronization.
4. Generate a mapping file with appropriate mapping rules.

> **See Also:** "Configuring Mapping Rules" on page 6-3 for instructions on creating mapping rules and for sample mapping files

Task 2: Configure Access Control Lists

Set up appropriate ACLs allowing read, add, or modify access rights on the subscribed domains.

During import operations, you would privilege the Oracle Internet Directory user `orclodipagentname=iPlanetImport,cn=subscriber profile,cn=changelog subscriber,cn=oracle internet directory` to update the subscribed domain in Oracle Internet Directory.

For example, assuming that no ACLs are applied to the domain of interest, the following LDIF sample can be used. In this file, the domain of interest is *`Synchronization_domain_in_OID`*.

```
ACL in OID:
dn: Synchronization_domain_in_OID
changetype: modify
add: orclaci
orclaci: access to entry by "orclodipagentname=iPlanetImport,cn=subscriber
profile,cn=changelog subscriber,cn=oracle internet directory" (browse,add,delete)
orclaci: access to attr=(*) by "orclodipagentname=iPlanetImport,cn=subscriber
profile,cn=changelog subscriber,cn=oracle internet directory"
(read,search,write,compare)"
```

On the other hand, the privileges can also be granted to the group `cn=odipgroup,cn=odi,cn=oracle internet directory` of which the profile is a member. However, remember that, when privileges are granted to the group, all members of the group are, intentionally or not, granted privileges.

During import operations, the user specified by the Connected Directory Account attribute in the integration profile must have read access to the change log and source container in the SunONE Directory Server. During export operations, the user specified by the Connected Directory Account attribute in the integration profile must have write access to the target container in the SunONE Directory Server.

> **See Also:** SunONE Directory Server documentation for instructions on how to apply ACLs on the SunONE Directory Server change log container and the SunONE Directory Server subscribed domain

Task 3: Prepare Both Directories for Synchronization

Follow these steps:

1. Before the start of the synchronization, make the data in the domains of interest to be equivalent. This can be achieved by the Directory Integration and Provisioning Assistant with the bootstrap option. Bootstrapping is described in Chapter 8, "Bootstrapping of a Directory in Oracle Directory Integration and Provisioning".
2. If you have used LDIF file-based bootstrapping, then you must initialize the `lastchangenumber` value. You can do this by using the Directory Integration and Provisioning Assistant:

   ```
   dipassistant mp -profile profile_name -updlcn
   ```

3. At the end of bootstrapping, be sure that the change logging option for the Oracle directory server is set to the default, namely, `TRUE`. If it is set to `FALSE`, then shut down the Oracle Internet Directory server and start with the change log enabled by using the **OID Control Utility**.

   Similarly, verify that change logging is enabled in SunONE Directory Server.

   See Also: The Oracle Internet Directory server administration tools chapter of the *Oracle Identity Management User Reference* for information on the OID Control

Task 4: (Optional) Configure the SunONE Directory Server External Authentication Plug-in

If you are storing passwords only in SunONE Directory Server and do not want to synchronize them with Oracle Internet Directory, then, to authenticate SunONE Directory Server users from Oracle Internet Directory, you must use the SunONE Directory Server external authentication plug-in.

This section tells how to install, delete, enable, and disable the SunONE Directory Server external authentication plug-in by using the command line. You can perform these operations, except for installation, by using Oracle Directory Manager as described in *Oracle Internet Directory Administrator's Guide*.

Note: The SunONE Directory Server external authentication plug-in can be configured to authenticate to only one single SunONE Directory Server.

Installing the SunONE Directory Server External Authentication Plug-in

To install the plug-in:

1. Execute `$ORACLE_HOME/ldap/admin/oidspipi.sh`.

   Note: To run shell script tools on the Windows operating system, you need one of the following UNIX emulation utilities:

   - Cygwin 1.3.2.2-1 or later. Visit: `http://sources.redhat.com`
   - MKS Toolkit 6.1. Visit: `http://www.datafocus.com/`

   To execute `oidspipi.sh`, enter:

   ```
   cd $ORACLE_HOME/ldap/admin
   oidspipi.sh
   ```

If you are using the Windows operating system, then execute oidspipi.sh after you have installed the UNIX emulation utility by entering:

```
sh oidspipi.sh
```

2. Enter the SunONE Directory Server host name. This is the SunONE Directory Server to which you are going to synchronize. This value is required.
3. Choose whether to use an SSL connection.

   When specifying the wallet location on the Microsoft Windows operating system, add an additional backslashes (\). For example, if the wallet location is `D:storage\wallet`, then enter `D:\\storage\\wallet`.
4. Enter the SunONE Directory Server port number.
5. Enter the database connect string.
6. Enter the ODS password. The default ODS password is the same as that set for the Oracle Application Server administrator during installation.
7. Enter Oracle directory server host name. This value is required.
8. Enter Oracle directory server port number. The default port is 389.
9. Enter the password of the Oracle administrator (`orcladmin`). This value is required.
10. Enter the distinguished name of the container to which the plug-in needs to be applied. Every entry in this container will be authenticated against SunONE Directory Server. Note that this need not necessarily be the User Search Base supplied in Oracle Internet Directory Self-Service Console. All the users under this search base are authenticated externally to the SunONE Directory Server. If more than one value is specified, then use semi-colons (;) to separate them.
11. Enter the Plug-in Request Group DN. For security reasons, the plug-in can be invoked only by users belonging to this group. For example, suppose that the Oracle Application Server Single Sign-On administrators are in the group `cn=OracleUserSecurityAdmins,cn=Groups,cn=OracleContext`. If you enter this value for the Plug-in Request Group DN, then only requests coming from Oracle Application Server Single Sign-On administrators can trigger the external authentication plug-in. You can enter multiple DN values. Use a semicolon (;) to separate them. This value is not required, but, for security purposes, it should be specified.
12. Enter the value of the entry that is to be excluded from authentication to SunONE Directory Server. This value is the exception to item 10 on page 20-9. You need to enter the value in the standard ldapsearch filter format. For example, if you specify the value `(&(objectclass=inetorgperson)(cn=orcladmin))`, then any entry under the user container specified in item 10 that has the `cn=orcladmin` and `objectclass=inetorgperson` attribute value will not be authenticated to SunONE Directory Server.
13. Specify whether you want to back up the SunONE Directory Server for failover.

Deleting the SunONE Directory External Authentication Plug-in

To delete the SunONE Directory Server plug-in by using Oracle Directory Manager, follow the instructions in the chapter on the Oracle Internet Directory plug-in framework in *Oracle Internet Directory Administrator's Guide*.

To delete the SunONE Directory Server plug-in by using command-line tools, use these commands:

```
ldapdelete -h host -p port -D cn=orcladmin -w password
"cn=ipwhencompare,cn=plugin,cn=subconfigsubentry"

ldapdelete -h host -p port -D cn=orcladmin -w password
"cn=ipwhenbind,cn=plugin,cn=subconfigsubentry"
```

Enabling the SunONE Directory External Authentication Plug-in

To enable the SunONE Directory external authentication plug-in by using Oracle Directory Manager, follow the instructions in the chapter on the Oracle Internet Directory plug-in framework in *Oracle Internet Directory Administrator's Guide*. Set the Plug-in Enable field to `1`.

To enable the SunONE Directory Server external authentication plug-in by using command-line tools, enter the following commands:

```
ldapmodify -h host_name -p port_number -D cn=orcladmin -w password <<EOF
dn: cn=ipwhencompare,cn=plugin,cn=subconfigsubentry
changetype: modify
replace: orclpluginenable
orclpluginenable: 1
EOF


ldapmodify -h host_name -p port_number -D cn=orcladmin -w password <<EOF
dn: cn=ipwhenbind,cn=plugin,cn=subconfigsubentry
changetype: modify
replace: orclpluginenable
orclpluginenable: 1
EOF
```

Disabling the SunONE Directory Server External Authentication Plug-in

To disable the SunONE Directory Server external authentication plug-in by using Oracle Directory Manager, follow the instructions in the chapter on the Oracle Internet Directory plug-in framework in *Oracle Internet Directory Administrator's Guide*. Set the Plug-in Enable field to `0`.

To disable the SunONE Directory Server external authentication plug-in by using command-line tools, enter the following commands:

```
ldapmodify -h host_name -p port_number -D cn=orcladmin -w password <<EOF
dn: cn=ipwhencompare,cn=plugin,cn=subconfigsubentry
changetype: modify
replace: orclpluginenable
orclpluginenable: 0
EOF


ldapmodify -h <host> -p <port> -D cn=orcladmin -w <password> <<EOF
dn: cn=ipwhenbind,cn=plugin,cn=subconfigsubentry
changetype: modify
replace: orclpluginenable
orclpluginenable: 0
EOF
```

Enabling and Disabling SunONE Directory External Authentication Plug-in Debugging

If you are experiencing unknown errors, the you can enable the plug-in debugging. To do this, enter:

```
sqlplus ods/odspassword @$ORACLE_HOME/ldap/admin/oidspdon.pls
```

To check the plug-in debugging log, enter:

```
sqlplus ods/ods
select * from plg_debug_log order by id;
```

To delete the plug-in debugging log, enter:

```
sqlplus ods/ods
truncate table plg_debug_log
```

To disable the plug-in debugging, enter:

```
sqlplus ods/ods @$ORACLE_HOME/ldap/admin/oidspdof.pls
```

Note: If you need to change the plug-in setup—that is, the information you entered in the installation steps—then you can rerun the installation script. Before you rerun the script, delete the SunONE Directory external authentication plug-in by following the instructions in "Deleting the SunONE Directory External Authentication Plug-in" on page 20-9.

See Also:

- The section on protection of user passwords for directory authentication in *Oracle Internet Directory Administrator's Guide* for a list of the hashing algorithms that Oracle Internet Directory supports for password protection
- SunONE Directory Server documentation for instructions on how to set the appropriate hashing algorithm for passwords in SunONE Directory Server

Task 5: Start the Synchronization

To start synchronization:

1. Enable the profile by setting the `profileStatus` attribute to `ENABLE` in either the Oracle Directory Integration and Provisioning Server Administration tool or the Directory Integration and Provisioning Assistant
2. Start the Oracle directory integration and provisioning server by using the OID Control Utility (oidctl) with the appropriate configuration set entry in which the profile is stored.

The Synchronization Process

The synchronization process is as follows:

1. In an import operation, the SunONE connector extracts all the changes from the SunONE Directory Server based on the value specified in the `orclodipConDirLastAppliedChgNum` attribute. It then applies them to Oracle Internet Directory.

   In an export operation, the SunONE connector extracts all the changes from Oracle Internet Directory based on the `orclodipLastAppliedChangeNumber` and applies them to the SunONE Directory Server.

2. Once all the changes are read and applied, the appropriate attribute—either `orclodipConDirLastAppliedChgNum` or `orclodipLastAppliedChangeNumber`—is updated.
3. After the execution is finished, the directory integration and provisioning server updates the execution status attributes.

Supported Configurations for Integrating with SunONE Directory Server

In a deployment with Oracle Internet Directory as the central directory, the following configurations are supported:

- Identical DITs on both directories
- Synchronization by using domain mapping
- Password synchronization. In this environment, synchronization ensures only the creation of footprints on the SunONE Directory Server. Any other configuration changes required to access the user or group entries must be specifically handled by the deployment.

In a deployment with SunONE Directory Server as the central repository, the following configurations are supported:

- Identical DITs on both directories
- Synchronization by using domain mapping
- Password synchronization
- Plug-in-based authentication from Oracle Internet Directory

Part VI

Appendixes

This part contains these appendixes:

- Appendix A, "Elements in the Oracle Directory Integration and Provisioning Server Administration Tool"
- Appendix B, "Case Study: A Deployment of Oracle Directory Integration and Provisioning"
- Appendix C, "Troubleshooting Oracle Directory Integration and Provisioning"

A

Elements in the Oracle Directory Integration and Provisioning Server Administration Tool

This appendix describes the tab pages and corresponding fields in the Oracle Directory Integration and Provisioning Server Administration tool. It contains these topics:

Windows and Fields for Connecting to a Directory Server

This section lists and describes the windows and fields you use to connect to a directory server.

Credentials

Table A–1 Fields in the Credentials Tab Page

User	The default value for the user name is `dipadmin`. This is the nickname of the user whose entry is `cn=dipadmin,cn=odi,cn=oracle internet directory`. If you have already set up the user's entry by using LDAP command-line tools, then you can enter that user's entry in one of two ways: ■ Browse and select that entry by using the button to the right of the User field ■ Type the **distinguished name (DN)** for that user entry by using the correct format, for example, `cn=Susie Brown,ou=HR,o=acme,c=us` If you do not have the correct privileges, then access to the tool is denied. To use this tool, you must be a member of the following group: `cn=dipadmingrp,cn=odi,cn=oracle internet directory`.

Table A–1 (Cont.) Fields in the Credentials Tab Page

Password	If you are logging in as the super user and you specified a password for the super user during installation, in the **Password** field, type the password you specified. Otherwise, type the default password, namely, `welcome`. After you are logged into Oracle Directory Integration and Provisioning Server Administration and have connected to a directory server, you should change this password to protect the directory. If you are logging in anonymously, leave the **Password** filed empty. If you want to login as a specific directory user, enter the corresponding password. **See Also:** The chapter on directory server administration in *Oracle Internet Directory Administrator's Guide*, for instructions on how to change the password
Server	The first time you log in, the Oracle Directory Integration and Provisioning Server Administration tool displays the name of default Oracle directory server you specified during the Oracle Application Server installation. It obtains the information for the directory server by checking first the value for the `oidhost` parameter in the `ias.properties` file in the $*ORACLE_HOME*/config directory. If no value is specified there, then it checks the value for the `host` parameter in the `osdadmin.ini` file. If no value is specifed there, then it displays the value `localhost` in the Server field. If you are want to connect to a server on a different host: 1. Click the button to the right of the **Server** list. The Select Directory Server dialog box displays a list of available servers. 2. Select a server. 3. Choose **OK**. To add a directory server to the list: 1. In the Select Directory Servers dialog box, choose **Add**. The Directory Server Connection dialog box appears. 2. In the **Server** field, type the name of the directory server you want to add. 3. In the **Port** field, type the port number for the server you want to add. 4. Choose **OK**. The added directory appears in the list in the Select Directory Server dialog box. To modify a directory server on the list: 1. Select the directory server you want to modify. 2. Choose **Edit**. The Directory Server Connection dialog box appears. 3. Modify the **Server** and **Port** fields, then choose **OK**. The modifications for that server appear in the list in the Select Directory Server dialog box.
Port	The first time you log in, the Oracle Directory Integration and Provisioning Server Administration tool displays the name of default Oracle directory server port you specified during the Oracle Application Server installation. It obtains this information by checking the value of the `oidport` parameter in the `ias.properties` file. If no value is specified there, then it checks the value for the `port` parameter in the `osdadmin.ini` file. If no value is specifed there, then it displays the value `389`. To change this port number: 1. Choose the button to the right of the **Server** field. 2. In the Select Directory Server dialog box, select the directory server. 3. Choose **Edit**. The Directory Server Connection dialog box appears. 4. In the Directory Server Connection dialog box, in the **Port** field, enter the new port number, then choose **OK**.

SSL

Table A–2 Fields in the SSL Tab Page

Field	Description
SSL Password	The password to open the user's wallet
SSL Authentication	Select the authentication level: ■ No SSL Authentication—Neither the client nor the server authenticates itself to the other. No certificates are sent or exchanged. If you selected the SSL Enabled check box on the Credentials tab, and choose this option, then only SSL encryption/decryption will be used. ■ SSL Client and Server Authentication—Two-way authentication. Both client and server send certificates to each other. ■ SSL Server Authentication—One-way authentication. Only the directory server authenticates itself to the client by sending its certificate to the client.

Configure Entry Management

Use this window to specify:

- The number of entries the Oracle Directory Integration and Provisioning Server Administration tool displays in a search result
- The duration of searches

You can make these configurations in either this tool or the directory server or both.

If you make the configuration in both this tool and the directory server, and the two configurations do not match, then Oracle Internet Directory resolves the conflict as follows:

- If the value you set in this tool is greater than that in the directory server, then the configuration of the server prevails. For example, if you set this tool to search for 2 minutes, and the directory server for 3 minutes, then the actual search duration will be 3 minutes.
- If the value you set in this tool is less than that in the directory server, then the configuration of this tool prevails. For example, if you set this tool to search for 2 minutes, and the server for 3 minutes, then the actual search duration is 2 minutes.

Configure Access Control Policy Management

Use this tab page to determine whether the navigator pane displays all ACPs automatically or only as the result of a search. If you have a large number of ACPs, then you may want to display them only as the result of a search.

Directory Server Connection

Use this dialog box to add a directory server to the list in the Select Directory Server dialog box.

Select Distinguished Name (DN) Path: Tree View

Use this dialog box to display the hierarchy of entries in the Directory Information Tree (DIT).

Click the plus sign (+) next to the top level entry to expand the tree. Expand the tree by clicking plus signs to see the subordinate entries. When you click a plus sign to expand an entry, that plus sign becomes a minus sign (-).

> **Note:** Although an entry that does not have subordinate entries may appear with a plus sign, when you click that plus sign, it disappears. Entries that have no plus or minus sign next to them are leaf nodes on the tree.

Select the entry you want and choose OK. That entry appears in the Root of the Search field in the Search window.

Select Directory Server

This dialog box displays a list of all directory servers to which you have connected at any time in the past. You can select a directory server from the list, either to connect to it, delete it, edit it, or to use it as a template for another management connection. To add a server to this list, choose **Add**. The Directory Server Connection dialog box appears.

Windows and Fields for Viewing Server Information

The windows and fields described in this section provide information about active server processes.

Active Processes

This window displays a list of currently active directory integration server instances. To display a configuration set entry in a format that is easier to read, select one of the entries and choose **View Properties**. To change the parameters, in the navigator pane, select the configuration set entry. The corresponding tab pages appear in the right pane.

Configuration Sets: Integration Profiles

This dialog box displays information about the directory integration profiles associated with a configuration set entry. If the Integration Profiles tab page is empty, then no directory integration profiles are associated with this configuration set entry. The columns of the Integration Profiles tab page in this dialog box are:

- Profile Name: The RDN component of the DN for this directory integration profile
- Synchronization Mode: Specifies whether the profile is used for importing or exporting. An import operation brings changes from a connected directory into Oracle Internet Directory. An export operation brings changes from Oracle Internet Directory into a connected directory.
- Profile Status: Specifies whether the profile is enabled or disabled

Windows and Fields for Registering and Editing a Directory Integration Profile

This section lists and describes the windows and fields you use when registering and editing a directory integration profile.

Integration Profiles

Use this dialog box to create or modify a directory integration profile. You can:

- Create an integration profile by copying an existing one. To do this, select the directory integration profile you want to copy, then choose Create Like. The Integration Profile dialog box displays the General tab page.
- Create an integration profile without copying an existing one. To do this, choose Create New. The Integration Profile dialog box displays the General tab page.
- Edit an integration profile by selecting it, then choosing Edit. This displays the General tab page.

General

Table A–3 Fields on the General Tab Page for Synchronization in the Oracle Directory Integration and Provisioning Server Administration Tool

Field	Description
Profile Name	Specify the name of the Profile. The name you enter is used as the RDN component of the DN for this integration profile. For example, specifying a profile name `MSAccess` creates an integration profile named `orclodipagentname=MSAccess,cn=subscriber profile, cn=changelog subscriber, cn=oracle internet directory`. This field is mandatory. There is no default.
Profile Version	Version of Oracle Directory Integration and Provisioning with which this profile was created.
Synchronization Mode	Specify whether this is an import or an export operation. An import operation pulls changes from a connected directory into Oracle Internet Directory. An export operation pushes changes from Oracle Internet Directory into a connected directory. This field is mandatory. The default is `IMPORT`.
Profile Status	Specify whether the profile is enabled or disabled. This field is mandatory. The default is `ENABLE`.
Profile Password	Specify the password that directory integration and provisioning server is to use when binding to Oracle Internet Directory on behalf of the profile. This field is mandatory and the default is `welcome`.
Scheduling Interval	Specify the number of seconds between synchronization attempts between a connected directory and Oracle Internet Directory. This field is mandatory. The default is `60`.
Maximum Number of Retries	Specify the maximum number of times the directory integration and provisioning server is to attempt synchronization before it disables synchronization. This field is mandatory. The default is 5. The first retry takes place 1 minute after the first failure. The second retry happens 2 minutes after the second failure, and subsequently the retry takes place n minutes after the n-th failure.
Debug Level	Specify the logging level for debugging as described in *Oracle Internet Directory Administrator's Guide*

Execution

Table A–4 Fields on the Execution Tab for Synchronization in the Oracle Directory Integration and Provisioning Server Administration Tool

Field	Description

Agent Execution Command	Specify the agent executable name and the arguments used by the directory integration and provisioning server to execute the agent.
	This field is optional. There is no default.
	A typical execution command is of the form,
	`odicmd user=%orclodipcondirAccessAccount pass=%orclodipcondiraccesspassword`
	Where `odicmd` is the command to be executed (available in the PATH or specified as a complete path name), and
	`user=%orclodipcondirAccessAccount pass=%orclodipcondiraccesspassword`
	are the command-line arguments. The value to be passed for the user is derived from the attribute `orclodipcondiraccessaccount`, and the value to be passed for `pass` is derived from the attribute `orclodipcondiraccesspassword`.
	A typical example is given in the Oracle Human Resources agent.

Connected Directory Account	Specify the account to be used by the connector/agent for accessing the connected directory. For example, if the connected directory is a database, then the account might be `Scott`. If the connected directory is another LDAP-compliant directory, then the account might be `cn=Directory Manager`. This field is optional. There is no default.
Connected Directory Account Password	Specify the password the connector/agent is to use when accessing the connected directory. This field is optional. There is no default.
Additional Config Info	This field displays additional information that the directory integration and provisioning server passes to an agent. You cannot modify this field through the Oracle Directory Integration and Provisioning Server Administration tool. The only way to modify it is to use Directory Integration and Provisioning Assistant.
Connected Directory URL	Connect details required to connect to the connected directory. This parameter refers to the host name and port number as `host:port:sslmode` To connect by using SSL, enter *`host:port`*:1. Make sure the certificate to connect to the directory is stored in the wallet, the location of which is specified in the file `odi.properties`. **Note:** To connect to SunONE Directory Server by using SSL, the server certificate needs to be loaded into the wallet. **See Also:** The chapter on Oracle Wallet Manager in *Oracle Advanced Security Administrator's Guide*
Interface Type	The format used by the import or export file. Options are `DB`, `LDAP`, `LDIF`, and `TAGGED`. This field is optional. The default is `TAGGED`.

Mapping

Table A–5 Fields on the Mapping Tab Page for Synchronization in the Oracle Directory Integration and Provisioning Server Administration Tool

Field	Description
Mapping Rules	This field displays the mapping rules for converting data between a connected directory and Oracle Internet Directory. There is no default. **Note:** You cannot edit the mapping rules file by using the Oracle Directory Integration and Provisioning Server Administration tool. You edit the mapping rules in a file manually and then upload it to the profile by using the Oracle Directory Integration and Provisioning.
Connected Directory Matching Filter	Specify the attribute that uniquely identifies an entry in the connected directory.
OID Matching Filter	Specify the attribute that uniquely identifies records in Oracle Internet Directory. This attribute is used as a key to synchronize Oracle Internet Directory and the connected directory. This field is optional.

Status

Table A–6 Fields on the Status Tab Page for Synchronization in the Oracle Directory Integration and Provisioning Server Administration Tool

Field	Description
OID Last Applied Change Number (Import operations only)	For export operations, specify the identifier of the last change from Oracle Internet Directory that has been applied to the connected directory. The default is `0`. The field can be consciously modified by the end user whenever appropriate. The profile should be in the disabled mode. If the number is increased, then any change log entries numbered between the original value and the new value will not be applied.
Last Execution Time	The most recent absolute time that the agent was executed. The default is the time at which the connector is created. Modifying this field will be misleading.
Last Successful Execution Time	The most recent absolute time that the agent succeeded. The default is the time at which the connector is created. Modifying this field will be misleading.
Synchronization Status	Synchronization success/failure.
Synchronization Errors	The last error message. You cannot modify this field. There is no default.
Last Applied Change Number (Export operations only)	The number of the change log entry that was most recently applied successfully to the connected directory. The field can be consciously modified by the end user whenever appropriate. The profile should be in the disabled mode. If the number is increased, then any change log entries numbered between the original value and the new value will not be applied.

Windows and Fields for Configuring the Active Directory Connector

This section describes the windows and fields you use when configuring the Active Directory Connector.

Active Directory Connector Express Synchronization Setup

Use this tab page to perform an express configuration of the Active Directory Connector. This configuration is based on an out-of-the-box installation of Oracle Application Server. Do not use this method to create any other type of directory integration profile.

Table A–7 Fields in the Active Directory Connector Express Synchronization Setup Tab Page

Field	Description
Active Directory Host	The host on which Microsoft Active Directory is installed
Active Directory Port	The port number for the Microsoft Active Directory installation
Account Name	The user name for logging into Microsoft Active Directory
Account Password	The password or logging into Microsoft Active Directory
Connector Name	The name of the directory integration profile
Import Profile Name	Read only. The value is derived from the profile of the connector
Export Profile Name	Read only. The value is derived from the profile of the connector
Configuration Set	The default is 1. If you specify another, then that configuration set is automatically created and associated with this profile.

You can also choose to specify access control policies.

B

Case Study: A Deployment of Oracle Directory Integration and Provisioning

This appendix describes a deployment in which Directory Integration and Provisioning integrates various applications in the MyCompany enterprise.

This section contains these topics:

Components in the MyCompany Enterprise

This hypothetical enterprise has the following components:

- Oracle Human Resources, in which all employees and contractors are managed
- An SunONE Directory Server, which is being used by certain applications
- An installation of OracleAS Portal, which is used as the intranet portal for all employees
- An installation of Oracle Content Management Software Development Kit, which is used as a document repository for all corporate documents

Requirements of the MyCompany Enterprise

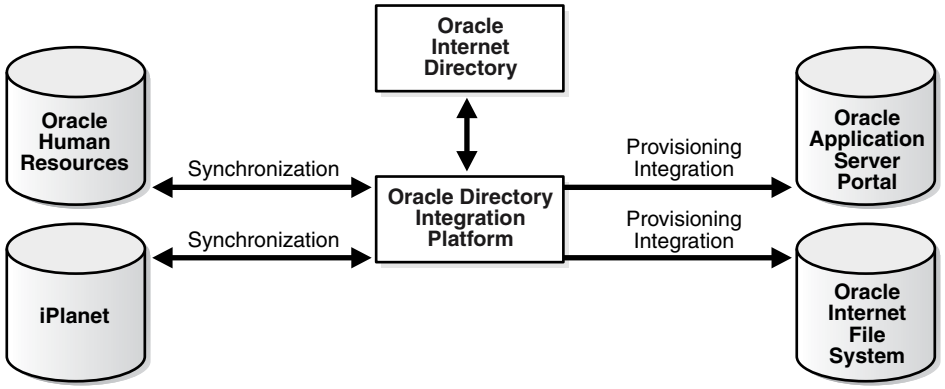
The MyCompany enterprise requires that:

- All employees and contractors are created in Oracle Human Resources. Once created, all applications in the enterprise must share this information through Oracle Internet Directory.
- All applications in the enterprise, including single sign-on services, can honor any employee created in Oracle Human Resources
- All applications interested in changes to user properties are notified when such changes occur
- A user's access rights are revoked when the user is terminated in Oracle Human Resources

Overall Deployment in the MyCompany Enterprise

Figure B–1 illustrates the various components and their relationships to each other.

Figure B–1 Example of Oracle Directory Integration and Provisioning in the MyCompany Deployment



In the example in Figure B–1:

- Oracle Internet Directory is the central user repository for all enterprise applications.
- Oracle Human Resources is the source of truth for all user-related information. It is synchronized with Oracle Internet Directory by using the Oracle Directory Synchronization Service.
- SunONE Directory Server, which is already deployed in the enterprise, is synchronized with Oracle Internet Directory by using the Oracle Directory Synchronization Service
- OracleAS Portal is notified of changes in Oracle Internet Directory by using the Oracle Provisioning Service
- Oracle Content Management Software Development Kit is notified of changes in Oracle Internet Directory by using the Oracle Provisioning Service.

User Creation and Provisioning in the MyCompany Enterprise

In this example, the MyCompany enterprise requires that all users be created in Oracle Human Resources. Directory Integration and Provisioning must propagate new user records to all other repositories in the enterprise.

Figure B–2 shows how Directory Integration and Provisioning performs this task.

Figure B–2 User Creation and Provisioning

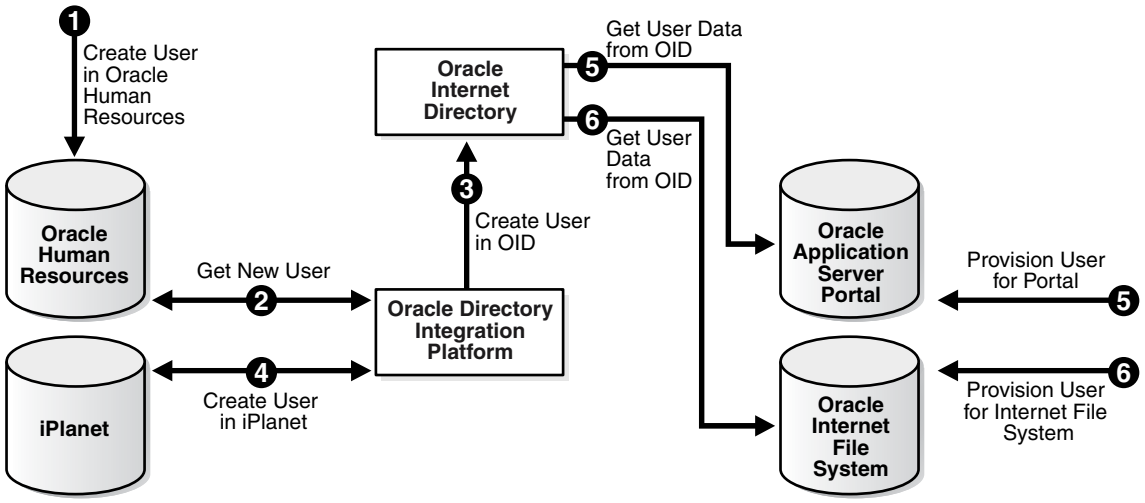


Figure B–2 shows the creation of a new user in Oracle Human Resources, which, in turn, causes an entry for that user to be created in Oracle Internet Directory and the SunONE Directory Server. It also shows the process of provisioning the user to access two applications in the enterprise: OracleAS Portal and Oracle Content Management Software Development Kit. User creation and provisioning occur in the following manner:

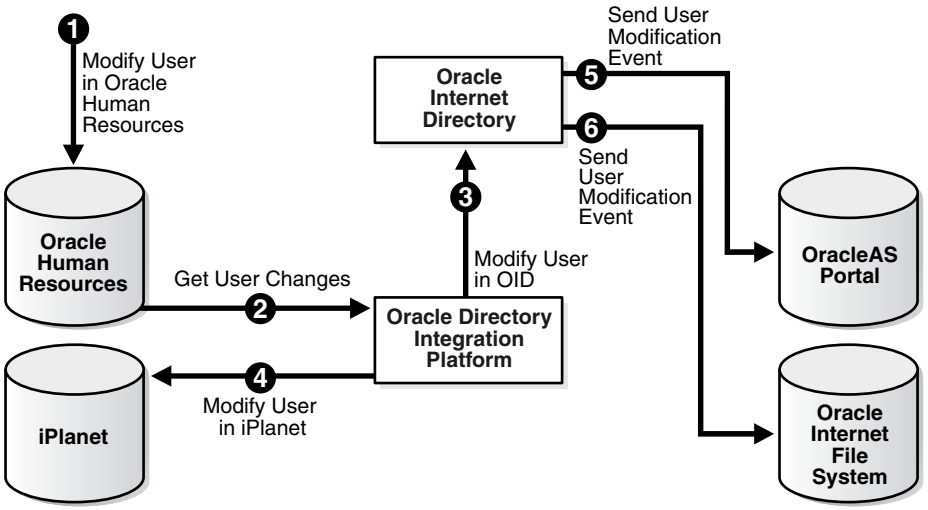
1. The Oracle Human Resources administrator creates the user in the Oracle Human Resources database.
2. Directory Integration and Provisioning, through the Oracle Directory Synchronization Service, detects the new-user creation.
3. Directory Integration and Provisioning, through the Oracle Directory Synchronization Service creates the entry for the user in Oracle Internet Directory.
4. Directory Integration and Provisioning, through the Oracle Directory Synchronization Service, creates an entry in the SunONE Directory Server.
5. Because the user entry is available in Oracle Internet Directory, the OracleAS Portal administrator can now provision the user to use the services of OracleAS Portal. During this task, the OracleAS Portal software automatically retrieves the user details from Oracle Internet Directory.
6. The Oracle Content Management Software Development Kit administrator also provisions the user to use Oracle Content Management Software Development Kit services by using a similar process.

Note that Directory Integration and Provisioning does not directly notify OracleAS Portal or Oracle Content Management Software Development Kit about new users. This is because not all users created in Oracle Human Resources need access to all services. In this case, the deployment must explicitly provision the users to use these services, as in steps 5 and 6.

Modification of User Properties in the MyCompany Enterprise

In this example, the MyCompany enterprise requires that any modification to user properties must be communicated to all components interested in such changes. Figure B–3 illustrates the actions that Directory Integration and Provisioning takes to meet this requirement.

Figure B–3 Modification of User Properties



The process is as follows:

1. The user is first modified in Oracle Human Resources.
2. Directory Integration and Provisioning retrieves these changes through the Oracle Directory Synchronization Service.
3. Directory Integration and Provisioning makes the corresponding user modification in Oracle Internet Directory.
4. The Oracle Directory Synchronization Service modifies the user in the SunONE Directory Server.
5. Directory Integration and Provisioning, through the Oracle Provisioning Service, notifies OracleAS Portal about the change in user properties.
6. Directory Integration and Provisioning, through the Oracle Provisioning Service, notifies Oracle Content Management Software Development Kit about the same change in user properties.

Deletion of Users in the MyCompany Enterprise

In this example, the MyCompany enterprise requires that a user being deleted or terminated in Oracle Human Resources be automatically denied access to all enterprise resources that are based on the directory service.

Figure B–4 shows the flow of events during the deletion of users:

Figure B–4 Deletion of Users from the Corporate Human Resources

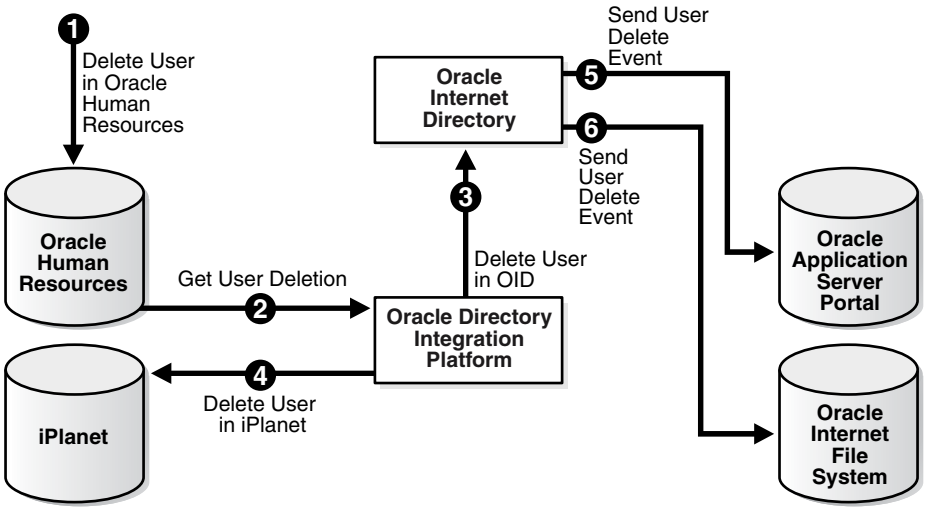


Figure B–4 shows the process by which Directory Integration and Provisioning communicates the deletion of users to all systems in the enterprise. The process is as follows:

1. The user is first deleted in the Oracle Human Resources.
2. Directory Integration and Provisioning retrieves these changes through the Oracle Directory Synchronization Service.
3. Directory Integration and Provisioning, through the Oracle Directory Synchronization Service, makes the corresponding user deletion in Oracle Internet Directory.
4. Directory Integration and Provisioning, through the Oracle Directory Synchronization Service, deletes the users in the SunONE Directory Server.
5. Directory Integration and Provisioning, through the Oracle Provisioning Service, notifies OracleAS Portal about the deletion of the user.
6. Directory Integration and Provisioning, through the Oracle Provisioning Service, notifies Oracle Content Management Software Development Kit about the deletion of the user.

Once all of the steps are completed, a deleted user in Oracle Human Resources can no longer access OracleAS Portal or Oracle Content Management Software Development Kit.

C

Troubleshooting Oracle Directory Integration and Provisioning

This appendix describes common problems that you might encounter when using Oracle Directory Integration and Provisioning and explains how to solve them. It contains the following topics:

See Also:

- Oracle by Example for Oracle Identity Management, available from the Oracle Technology Network at `http://www.oracle.com/technology/index.html`
- *Oracle Identity Management User Reference*

Diagnosing Oracle Directory Integration and Provisioning Server Problems

This section includes general approaches for diagnosing problems with the Oracle directory integration and provisioning server. It contains the following topics:

Diagnosing the Oracle Directory Integration and Provisioning Server in an Infrastructure Installation

After you start the Oracle directory integration and provisioning server, you can verify that it is running by following these steps:

1. On UNIX, use the following command to verify that `odisrv` process is running:

   ```
   ps -ef | grep odisrv
   ```

   For Windows operating systems, obtain the value of process ID (PID) for the `odisrv` process from *$ORACLE_HOME*/ldap/log/oidmon.log. Then, launch Task Manager and click the Processes tab to verify that the process is running.

2. If the Oracle directory integration and provisioning server is not running, then examine the *$ORACLE_HOME*/ldap/log/oidmon.log file to determine the reason why the server did not start.

3. If the log file shows any database related errors:

   a. Verify that a value is set for `ORACLE_SID`.

   b. Verify that the connect string assigned to `ORACLE_SID` is specified in the *$ORACLE_HOME*/network/admin/tnsnames.ora file.

4. Ensure that the log file lists valid values for the server `instance` number and the `configset` number arguments. If the values are set correctly, then examine the file *$ORACLE_HOME*/ldap/log/odisrv_*xx*.log where `xx` is the number of the started instance. If the odisrv_*xx*.log file indicates a registration error, then re-register the Oracle directory integration and provisioning server by using `odisrvreg`.

5. If you do not find any errors in the previous step, then examine the file *$ORACLE_HOME*/ldap/log/odisrv_jvm_*yyy*.log, where *yyy* is the process identifier of the `odisrv` process that should have started. Look for the file with the latest timestamp.

Diagnosing the Oracle Directory Integration and Provisioning Server in an Oracle Directory Integration and Provisioning-Only Installation

After you start the Oracle directory integration and provisioning server, you can verify that it is running by following these steps:

1. On UNIX, use the following command to verify that `odisrv` process is running:

   ```
   ps -ef | grep odisrv
   ```

   For Windows operating systems, obtain the value of process ID (PID) for the `odisrv` process from the *$ORACLE_HOME*/ldap/log/odisrv_*xx*.log file, where `xx` is the number of the started instance. Then, launch Task Manager and click the Processes tab to verify that the process is running.

2. If the Oracle directory integration and provisioning server is not running, examine the odisrv_*xx*.log file. If the file contains a registration error, then re-register the Oracle directory integration and provisioning server by using `odisrvreg`.

3. If you do not find any errors in the previous step, then examine the file *$ORACLE_HOME*/ldap/log/odisrv_jvm_*yyy*.log, where *yyy* is the process identifier of the `odisrv` process that should have started. Look for the file with the latest timestamp.

Troubleshooting Utilities

This section discusses the `oditest` and DIP Tester utilities that you can use to troubleshoot synchronization problems.

The oditest Utility

Troubleshooting synchronization can be complex if there are numerous profiles running or if the synchronization interval for a particular profile is set to occur too infrequently. In such cases, the behavior of any connector can be tested using the `oditest` utility as follows:

1. If numerous profiles are running, then use the Directory Integration and Provisioning Assistant to selectively disable the profile you want to troubleshoot. If a single profile is running, then stop the directory integration and provisioning server.
2. Go to *$ORACLE_HOME*/bin and run the `oditest` utility using the following syntax:

   ```
   oditest sync | prov profile_name host=host_of_Oracle_Internet_Directory \
   port=port_for_Oracle_Internet_Directory binddn=bind_DN \
   bindpass=password_for_the_bind_DN sslauth=0 debug=63
   ```

   The following example shows how to run the `oditest` utility with a SunONE Directory Server synchronization profile:

   ```
   oditest sync IplanetImport host=my-oidhost port=3060 binddn=cn=orcladmin
   bindpass=welcome1 sslauth=0 debug=63
   ```

   See Also: The chapter on logging, auditing, and debugging the directory in *Oracle Internet Directory Administrator's Guide*

The DIP Tester Utility

The DIP Tester utility is a standalone, platform independent Java application that aids in the configuration, testing, and debugging of Oracle Internet Directory implementations that synchronize with SunONE (iPlanet) Directory Server or Microsoft Active Directory. The utility uses the Directory Integration and Provisioning Assistant (`dipassistant`) to modify profiles and also uses standard LDAP tools (`ldapadd`, `ldapmodify`, `ldapdelete`, and `ldapsearch`) for many behind-the-scenes operations. The DIP Tester utility has been tested on Oracle Internet Directory Release 10*g* (9.0.4) through Oracle Application Server 10*g* Release 2 (10.1.2) for Solaris, Linux, and Windows platforms. You can download DIP Tester from Oracle Technology Network at `http://www.oracle.com/technology/index.html`. The download includes graphical user interface (GUI) and command-line versions of the DIP Tester utility. Both versions are installed automatically with a single install script.

As you follow the troubleshooting procedure in this section, you can use DIP Tester to:

- Make changes to a directory integration profile
- View log files
- Create test entries
- Get or set the last applied change key
- Dump entire profile contents
- Reload the map file
- Start and stop the directory integration and provisioning server
- Capture errors in trace files for uploading to Oracle Support
- Perform initial bootstrapping of users

Note: When the directory integration and provisioning server performs a synchronization, it reads the last applied change key and caches the value. At the next synchronization interval, the directory integration and provisioning server updates Oracle Internet Directory with the last execution time and the cached value of the last applied change key.

Before you manually change the last applied change key in a synchronization profile, be sure to stop the directory integration and provisioning server. Otherwise at the next interval your change will be overwritten by the cached value. In fact, you should always stop the directory integration and provisioning server before changing any values in a synchronization profile.

DIP Tester is installed in the *$ORACLE_HOME*/bin directory.

See Also: The README.txt and DIP Tester User's Guide, located in the directory where you installed the DIP Tester utility

Problems and Solutions

This section describes common problems and solutions for Oracle Directory Integration and Provisioning. It contains the following topics:

- Oracle Directory Integration and Provisioning Server Errors
- Provisioning Errors and Problems
- Synchronization Errors and Problems
- Windows Native Authentication Error and Problems
- Microsoft Active Directory and SunONE Directory Server Synchronization Errors and Problems

Note: The Oracle directory integration and provisioning server stores error messages in the appropriate file, as described in "Location and Naming of Files" on page 6-14.

Oracle Directory Integration and Provisioning Server Errors

This section provides solutions for errors and problems you may encounter with the Oracle directory integration and provisioning server.

Problem

PASSWORD POLICY ERROR :9000: GSL_PWDEXPIRED_EXCP.

Solution

Beginning with Oracle Internet Directory 10*g* (9.0.4), the default password expiry time, which is assigned to the `pwdmaxage` attribute, is set to 60 days. To fix this problem, perform the following steps:

1. You must first unlock the `cn=orcladmin` super user account before you can modify password policies. Use the `oidpasswd` utility to unlock the super user account as follows:

```
oidpasswd connect=asdb unlock_su_acct=true
OID DB user password:
OID super user account unlocked successfully.
```

This unlocks only the super user account, `cn=orcladmin`. Do not confuse this account with the cd=orcladmin account within the default realm `cn=orcladmin,cn=users,dc=xxxxx,dc=yyyyy`. They are two separate accounts.

2. Launch an Oracle Internet Directory 10*g* (10.1.2) version of Oracle Directory Manager and navigate to Password Policy Management. You will see two entries: `cn=PwdPolicyEntry` and the password policy for your realm—for example, *`password_policy_entry`*`,dc=acme,dc=com`.

   Change the `pwdmaxage` attribute in each password policy to an appropriate value:

   - 5184000 = 60 days (default)
   - 7776000 = 90 days
   - 10368000 = 120 days
   - 15552000 = 180 days
   - 31536000 = 1 year

   Note: It is very important to change this value in both places.

3. Launch the Oracle Directory Manager and navigate to the realm-specific `orcladmin` account. Find the `userpassword` attribute and assign a new value. You should then be able to launch any Oracle component that uses OracleAS Single Sign-On and log in as `orcladmin`.

4. Rerun the `odisrvreg` utility to reset the randomly generated password for Directory Integration and Provisioning:

```
odisrvreg -D cn=orcladmin -w welcome1 -p 3060
Already Registered...Updating DIS password...
DIS registration successful.
```

Provisioning Errors and Problems

This section provides solutions for provisioning errors and problems.

Problem

Unable to get the Entry from its GUID. Fatal Error...

Solution

The Oracle directory integration and provisioning server is attempting to retrieve an entry that has been deleted, but not yet purged. Update the tombstone purge configuration settings in the Garbage Collection Management node of Oracle Directory Manager.

Problem

LDAP connection failure.

Solution

Directory Integration and Provisioning failed to connect to the directory server. Check the connection to the directory server.

> **See Also:** The chapter on directory server administration in *Oracle Internet Directory Administrator's Guide* for information about directory server connections

Problem

LDAP authentication failure.

Solution

The provisioning profile is not able to connect to the LDAP server as administrator. Verify Oracle directory integration and provisioning server entry in the directory. Re-register the Oracle directory integration and provisioning server by using `odisrvreg`.

> **See Also:** "Manually Registering the Oracle Directory Integration and Provisioning Server" on page 4-14

Problem

Initialization failure.

Solution

Problem in connecting to the directory server using JNDI. Examine the trace/audit file in *$ORACLE_HOME*/ldap/odi/log/*profile_name*.trc.

Problem

Database connection failure.

Solution

Problem connecting to the database with the given account information; either the database is not running or there is an authentication problem. Examine the trace/audit file in *$ORACLE_HOME*/ldap/odi/log/*profile_name*.trc.

Problem

Exception while calling SQL operation.

Solution

Problem in executing the package. Verify the package usability. Examine the trace/audit file in *$ORACLE_HOME*/ldap/odi/log/*profile_name*.trc.

Problem

Provisioning Profiles Not Getting Executed by the DIP Provisioning Server.

Solution

Provisioning profiles only execute when the Oracle directory integration and provisioning server is started with configuration set 0. Ensure that the Oracle directory integration and provisioning server has been started with the argument `configset=0`.

Problem

Unable to Connect to the Application Database.

Solution

The application database connection requirements in a provisioning profile may be incorrect. Use `sqlplus` to verify connectivity requirements.

Problem

USER/GROUP MODIFY and DELETE Events Not being consumed by the application.

Solution

The Oracle Provisioning Service first queries an application database about the existence of a user or group. If the application database responds with a negative value, then the user or group does not exist, and the event is not propagated to the application. Examine the trace/audit file in *$ORACLE_HOME*/ldap/odi/log/*profile_name*.trc to determine whether the user or group exists in the application database.

Problem

Subscription to Binary Attributes results in the Event propagation error.

Solution

Binary attributes propagation is not supported. Remove the binary attribute assignments from the event subscription in the provisioning profile.

Problem

Insufficient Access Rights to do "proxy" as the Application DN.

Solution

The Oracle Directory Integration and Provisioning server group has not been granted browse privilege by the application DN. Use the `ldapmodify` command to load the following ACIs, which grant browse privileges from the application DN to the Oracle Directory Integration and Provisioning group:

```
orclaci: access to attr=(*) by group="cn=odisgroup,cn=odi,cn=oracle internet
directory"(read,write,search,compare)
orclaci: access to entry by group="cn=odisgroup,cn=odi,cn=oracle internet
directory"(browse,proxy)
```

Problem

Insufficient access rights to use an application DN as proxy.

Solution

The Oracle Directory Integration and Provisioning server group has not been granted proxy privileges by the application DN. Use the `ldapmodify` command to load the following ACI, which grants proxy privileges from the application DN to the Oracle Directory Integration and Provisioning group:

```
orclaci: access to entry by group=" cn=odisgroup, cn=odi,cn=oracle internet
directory" (browse,proxy)
```

Synchronization Errors and Problems

This section provides solutions for synchronization errors and problems.

See Also: MetaLink Note: 276481.1—Troubleshooting OID DIP Synchronization Issues available on Oracle MetaLink at http://metalink.oracle.com/

Problem

LDAP: error code 50 - Insufficient Access Rights; remaining name 'CN=Users,dc=mycompany,dc=com'

Solution

The record target is not in a default container. Find the DST CHANGE RECORD. Check the ACIs for the target container. If they are blank, then use DIP Tester to apply a known set of ACIs to the new container.

Problem

LDAP: error code 50 - Insufficient Access Rights; ACTIVECHGIMP MAPPING IMPORT OPERATION FAILURE; Agent execution successful, Mapping/import operation failure

Solution

By default the `cn=Users,<default realm>` contains the proper ACIs. However, this error can occur when trying to synchronize into a different container within the default realm. Open the trace file, locate the change record that is causing the error, and then check the ACIs for the record's parent container. Apply the same ACIs to the target container.

Problem

Trace File Error: Not able to construct DN Output ChangeRecord ChangeRecord : Changetype: 1 ChangeKey: cn=users, dc=us,dc=oracle,dc=com Exception javax.naming. ContextNotEmptyException: [LDAP: error code 66 - Not Allowed On Non-leaf]; remaining name 'cn=users,dc=us,dc=oracle,dc=com' Missing mandatory attribute(s).

Solution

Problem with the mapping file. Follow the instructions in Oracle MetaLink Note: 261342.1—Understanding DIP Mapping available on Oracle MetaLink at http://metalink.oracle.com/.

Problem

Trace File Error: IPlanetImport:Error in Mapping Enginejava.lang.NullPointerException java.lang.NullPointerException at oracle.ldap.odip.engine.Connector.setValues(Connector.java:101).

Solution 1

The mapping file has not been loaded. In the Oracle Directory Integration and Provisioning Server Administration tool, verify that the Mapping tab contains the values from your mapping file. If your values are not available, then use DIP Tester to reload the mapping file.

Solution 2

The `orclcondirlastappliedchgnum` attribute is null or has no value. This may occur if bootstrapping failed or if you manually populated Oracle Internet Directory and did not assign a value to the `orclcondirlastappliedchgnum` attribute. Verify

that the orclcondirlastappliedchgnum attribute has a value. If not, then use DIP Tester to set the orclcondirlastappliedchgnum attribute.

Problem

Trace File Error: Command exec successful IPlanetImport:Error in Mapping Enginejava.lang.NullPointerException java.lang.NullPointerException at oracle.ldap.odip.engine.Connector.setValues(Connector.java:101) at oracle.ldap.odip.gsi.LDAPReader.initialise(LDAPReader.java:169) Updated Attributes orclodipLastExecutionTime: 20040601143204.

Solution

Missing LDAP port on connected directory URL attribute value (hostname:port). Specify the LDAP port in the connected directory URL attribute.

Problem

Trace File Error: LDAP URL : (xxxxxx.com:389<login credentials to 3rd party ldap server> LDAP Connection success ActiveChgImp:Error in Mapping EngineODIException: DIP_GEN_INITIALIZATION_EXCEPTION ODIException: DIP_GEN_INITIALIZATION_EXCEPTION at oracle.ldap.odip.util.DirUtils.getLastChgNum(DirUtils.java:48) at oracle.ldap.odip.gsi.LDAPReader.initAvailableChgKey(LDAPReader.java:719) at oracle.ldap.odip.gsi.LDAPReader.initialise(LDAPReader.java:212) at oracle.ldap.odip.engine.AgentThread.mapInitialise(AgentThread.java:327) at oracle.ldap.odip.engine.AgentThread.execMapping(AgentThread.java:253) at oracle.ldap.odip.engine.AgentThread.run(AgentThread.java:149) ActiveChgImp:about to Update exec status Error in proxy connection : java.lang.NullPointerException.

Solution

Permissions and ownership of the files in *$ORACLE_HOME*/ldap/odi/conf should be owned by Oracle installer id. Use ldapmodify to fix the following two entries:

```
dn: orclODIPAgentName=profile_name,cn=subscriber profile,
  cn=changelog subscriber, cn=oracle internet directory
changetype: modify
replace: orclaci
orclaci: access to attr = (*) by group="cn=odisgroup,cn=odi,cn=oracle
  internet directory"  (read,write,search,compare)
orclaci: access to entry by group="cn=odisgroup,cn=odi,cn=oracle
  internet directory"  (browse,proxy)

dn: orclodipAgentName=ActiveChgImp,cn=subscriber profile,cn=changelog
subscriber,cn=oracle internet directory
orclodipagentconfiginfo:: W0lOVEVSRkFDRURFVEFJTFNdClBhY2thZ2U6IGdzaQpSZWFkZXI
6IEFjdGl2ZUNoZ1JlYWRlcgo=
```

Note: The preceding entry is a binary object representing an import profile for the ActiveChange Reader. If you are fixing an SunONE/iPlanet, or and EXPORT profile, then you must dump the orclodipagentconfiginfo attribute for the corresponding profile from a existing profile or another node.

See Also: The following for information about LDAP error code 49 and Error 9000: GSL_PWDEXPIRED_EXCP:

- "Oracle Directory Integration and Provisioning Server Errors" on page C-4
- Oracle MetaLink Note: 265397.1—Password Policy Expires, available on Oracle MetaLink at `http://metalink.oracle.com/`

Problem

Mapping tab in the Oracle Directory Integration and Provisioning Server Administration tool shows file name instead of mapping rules.

Solution

The absolute path was not included when the mapping file was loaded. Reload the map file using full absolute path. You can reload the map file using the Directory Integration and Provisioning Assistant (`dipassistant`) or DIP Tester.

Windows Native Authentication Error and Problems

This section provides solutions for errors and problems you may encounter when integrating Oracle Identity Management with Windows Native Authentication.

Problem

Internal Server error. Please contact your administrator.

Solution

Windows native authentication is misconfigured on the middle tier computer. To fix this problem, perform the following steps:

1. Check the opmn.log file for errors.
2. Check ssoServer.log for errors.
3. Make sure that the keytab file is located in the $*ORACLE_HOME*/j2ee/OC4J_SECURITY/config directory and that the principal name configured in jazn-data.xml is correct.
4. Make sure that the single sign-on middle tier computer is properly configured to access the Key Distribution Center. See "Set Up a Kerberos Service Account for the OracleAS Single Sign-On Server" on page 18-41.

Problem

Could not authenticate to KDC.

Solution

This error message may be invoked if the realm name in krb5.conf is incorrectly configured. Check the values `default_realm` and `domain_realm` in /etc/krb5/krb5.conf. Note that the realm name is case sensitive.

Problem

Your browser does not support the Windows Kerberos authentication or is not configured properly.

Solution

The user's Web browser is not supported or is misconfigured. Follow the instructions in "Task 6: Configure Internet Explorer for Windows Native Authentication" on page 18-43.

Problem

"Access forbidden" or "HTTP error code 403" or "Windows Native Authentication Failed. Please contact your administrator."

Solution

These error messages have the same cause: the user entry cannot be found in Oracle Internet Directory. A local administrator working at a Windows desktop may be trying to access a single sign-on partner application whose entry may not have been synchronized with Oracle Internet Directory. Determine whether the user entry exists in the directory and if the Kerberos principal attributes for the user are properly synchronized from Microsoft Active Directory.

Problem

The windows login dialog box (with username, password, and domain fields in it) comes up when accessing the partner application.

Solution

The single sign-on server was not able to authenticate the Kerberos token because the corresponding user entry could not be found in Oracle Internet Directory. Add the user entry to the directory.

Problem

Single sign-on server fails to start. Log file contains an exception bearing the message "Credential not found."

Solution

The parameter `kerberos-servicename` may not be configured correctly. To fix this problem, perform the following steps:

1. Make sure that `kerberos-servicename` is configured correctly in the files orion-application.xml and jazn-data.xml. In orion-application.xml, the format for this parameter is `HTTP@sso.mycompany.com`. In the jazn-data.xml, the format is `HTTP/sso.mycompany.com`.
2. Check ssoServer.log for errors.
3. Make sure that the keytab file is located in the $*ORACLE_HOME*/j2ee/OC4J_SECURITY/config directory and that the principal name configured in jazn-data.xml is correct.
4. Make sure that the single sign-on middle tier computer is configured to access the Kerberos domain controller. See "Set Up a Kerberos Service Account for the OracleAS Single Sign-On Server" on page 18-41.

Microsoft Active Directory and SunONE Directory Server Synchronization Errors and Problems

This section provides solutions to synchronization errors and problems that can occur with Microsoft Active Directory and SunONE Directory Server.

Problem

LDAP: error code 50 - Insufficient Access Rights.

Solution

The odi agent `orclODIPAgentName=IPlanetImport,cn=subscriber profile,cn= changelog subscriber,cn=oracle internet directory` does not have full read/write access to the synchronized entries in Oracle Internet Directory. Because the `cn=oracleDASCreateUser,cn=groups,cn=oraclecontext,identity_management_realm` group will already have the required ACLs defined, this entry should be a member of this group. In this case, <subscriber DN> is set to *`identity_management_realm`*. You must add the `orclODIPAgentName=IPlanetImport,cn=subscriber profile,cn=changelog subscriber,cn=oracle internet directory` user entry to the `cn=oracleDASCreateUser,cn=groups,cn=oraclecontext,identity_management_realm` group, so that it will have the required ACL access to perform the updates: In Oracle Directory Manager, navigate through: Entry Management ->dc=com,*`identity_management_realm`*,cn=oraclecontext-> cn=groups-> cn=oracleDASCreateUser. From here, against the attribute 'uniquemember' add: orclODIPAgentName=IPlanetImport,cn=subscriber profile,cn=changelog subscriber,cn=oracle internet directory.

Problem

Add and change operations are successful, but delete operations fail without being recorded in the trace file.

Solution 1

In SunONE/iPlanet: Tombstones are not enabled. Verify that tombstones are enabled as described in Oracle MetaLink Note: 219835.1, available on Oracle MetaLink at `http://metalink.oracle.com/`.

Solution 2

In Microsoft Active Directory: The account used for the profile is not a member of the DIR SYNCH ADMIN group. This only occurs if you are not using a Microsoft Active Directory administrator account. Install the appropriate patch from Microsoft.

Problem

Data synchronization problems encountered after configuring Oracle Directory Integration import or export connectors to third-party LDAP directories.

Solution

Determine the cause by running the `oditest` utility. Run the `oditest` utility as described in Troubleshooting Integration with the SunONE Connector on page C-26 or Debugging the Active Directory Connector on page C-25.

Problem

The Oracle Internet Directory profile in Oracle Directory Manager shows "synchronization successful" yet no changes show up in the directory.

Solution

The synchronization interval is set to occur too infrequently to be of use during testing. By default, the synchronization interval is set to occur every 60 seconds. However, you may increase the synchronization interval for better performance. For example, you may increase your synchronization interval to a value such as 300 seconds (5 minutes) or 600 seconds (10 minutes). Follow these steps to decrease your synchronization interval:

> **WARNING: Decreasing your synchronization interval may significantly impact the performance of your connected directory server. Before changing your synchronization interval, try debugging your connector with the `oditest` utility. If you do change your synchronization interval, be sure to reset it to its original value once you are finished with your testing procedures.**

1. In the Oracle Directory Integration and Provisioning Server Administration tool, in the navigator pane, navigate to the Integration Server and modify the Scheduling Interval attribute in the profiles to 20 seconds.
2. Use the `odisrv` command to stop the directory integration and provisioning server and restart it with the parameter `debug=63`.
3. Add a test entry in your connected directory.
4. In Oracle Internet Directory, change to the *$ORACLE_HOME*/ldap/odi/log directory and use the `cat` command to display the file ActiveChgImp.trc. When the directory integration and provisioning server wakes up and processes the record from the connected directory changelog, you will see the details listed in the IplanetImport.trc or ActiveChgImp.trc file.
5. Examine the trace files for possible clues as to what is actually taking place: You should see the handshake/login to the connected directory server, then the change being captured and reformatted according to the mapping rules, and finally the change being attempted in Oracle Internet Directory. If there are handshake or mapping problems they will appear in this file.

A common mistake is to set the Connect Directory Account DN to Administrator. This field must contain the entire distinguished name of the Active Directory administrator—for example:

```
cn=Administrator,cn=Users,dc=myoracle,dc=com
```

The first domain component is the value of the third field of the Windows Login Page: User Name, Password, Log on to.

The following `ldapsearch` commands may be helpful in identifying problems with the configuration.

To check the default identity management realm:

```
ldapsearch -h host -p port -D cn=orcladmin -w password -b "cn=common,cn=products,
cn=oraclecontext" -L -s
base "objectclass=*" orcldefaultsubscriber
```

To dump the directory integration and provisioning server configuration set:

```
ldapsearch -h host -p port -D cn=orcladmin -w password -b cn=instance1,cn=odisrv,
cn=subregistrysubentry
-s base -v "objectclass=*"
```

To check profiles:

```
ldapsearch -h host -p port -D cn=orcladmin -w password -b
"orclODIPAgentName=profile,cn=subscriber profile,cn=changelog Subscriber,cn=oracle
internet directory" -s sub objectclass=*
```

To check the agent credentials:

Note: This command returns the password in clear text only if you run it using `orcladmin` credentials.

```
ldapsearch -p port -D cn=orcladmin -w password -b "orclODIPAgentName=profile,
cn=subscriber profile,cn=changelog subscriber,cn=oracle internet directory"
-s sub "objectclass=*"
```

Problem

Bootstrap Error: DIP_GEN_AUTHENTICATION_FAILURE when trying to Synchronize Active Directory with Oracle Internet Directory

Solution

Invalid credentials. Check the synchronization profile and ensure that it contains the proper credentials to log in to the Active Directory server.

Troubleshooting Provisioning

This section describes how to troubleshoot provisioning problems in the Oracle Internet Directory Provisioning Console. It contains these topics:

Viewing Diagnostic Settings

You can use the Oracle Delegated Administration Services diagnostic settings to debug provisioning problems in the Oracle Internet Directory Provisioning Console without having to examine the log files. For more information on viewing and configuring diagnostic settings, see the chapter on managing users and groups with the Oracle Internet Directory Self-Service Console in the *Oracle Identity Management Guide to Delegated Administration*.

Provisioning-Integration Applications Not Visible in the Provisioning Console

After you install a new provisioning-integrated application in Oracle Internet Directory, the application does not appear in the Provisioning Console until you reload the application cache. You must also reload the application cache whenever a provisioning-integrated application is enabled or disabled in Oracle Internet Directory.

To reload the application cache, follow the procedures described in "Reloading the Application Cache" on page 14-5.

Unable to Create Users

The Oracle Provisioning Service uses plug-ins to create new users. This section contains the following topics, which describe how to troubleshoot the Oracle Provisioning Service plug-ins to resolve user creation problems:

- Troubleshooting Data Entry Plug-Ins
- Troubleshooting Provisioning Plug-Ins

Troubleshooting Data Entry Plug-Ins

Provisioning-integrated applications can invoke the Pre-Data Entry and Post-Data Entry plug-ins to enhance provisioning intelligence and implement business policies. This section describes how to troubleshoot problems with both plug-ins.

Identifying Problems with the Pre-Data Entry Plug-In When you follow the instructions described in "Creating Users with the Provisioning Console" on page 14-2, the Provisioning Console invokes the Pre-Data Entry plug-in after you click the Next button in the General Provisioning window. The primary purpose of this plug-in is to determine whether a user should be provisioned in the applications selected in the General Provisioning window. If a user has provisioning permission for an application, then the Pre-Data Entry plug-in populates fields in the next window, the Application Provisioning window, according to the application's provisioning policies.

In the event of a problem with the Pre-Data Entry plug-in, an error containing an exception message and stack trace will display in the General Provisioning window. You can find the user attributes that were passed to the plug-in by locating the following line in the stack trace:

```
******preplugin base user prop set for <Application Name> …
```

You can locate the error in the log files by searching for the following:

```
oracle.idm.provisioning.plugin.PluginException
```

Identifying Problems with the Post-Data Entry Plug-In When you follow the instructions described in "Creating Users with the Provisioning Console" on page 14-2, the Provisioning Console invokes the Pre-Data Entry plug-in after you click the Next button in the Application Attributes window. The Post-Data Entry plug-in validates data entered by users for common and application-specific attributes. The validation for the plug-in must be successful in order for provisioning to continue.

In the event of a problem with the Post-Data Entry plug-in, an error will display in the Application Attributes window. The exception stack trace will be located after the following line:

```
UserPlguInMgmt::postPlugInProcess(): apptype <Application Type> appname
<Application Name> error when executing plugin logics
```

Troubleshooting Provisioning Plug-Ins

Provisioning-integrated applications can be provisioned either through a PL/SQL plug-in or the Data Access Java plug-in. The PL/SQL plug-in is invoked by the Oracle directory integration and provisioning server while the Data Access Java plug-in is invoked directly by Oracle Delegated Administration Services.

When you follow the instructions described in "Creating Users with the Provisioning Console" on page 14-2, user creation may be successful even though provisioning for a specific application may fail. You will know when provisioning has failed if you receive a warning status along with a provisioning error message after you click the Submit button in the Review window. For details on the failure, search the log files for "Data Access plug-in execution failure". The lines following this statement list details of why provisioning failed.

Using Provisioning Status to Identify Problems

You can use the provisioning status of a user entry to help identify provisioning problems.

To view a user entry's provisioning status:

1. In the Provisioning Console, select the **Directory** tab, then select **Users**. The Search for Users window appears.
2. In the **Search for User** field, enter the first few characters of the user's first name, last name, e-mail address, or user ID. For example, if you are searching for Anne Smith, you could enter `Ann` or `Smi`. To generate a list of all users in the directory, leave this field blank.
3. Choose **Go** to display the search results.
4. Select the user whose entry you want to view, then click the **View** button to display the View User window.

   This window is described in *Oracle Identity Management Guide to Delegated Administration*
5. In the **View User** window, examine the entries in the **Provisioning Status** table. If the Provisioning Status column for an application contains a value of `PROVISIONING_FAILURE`, then the Provisioning Status Description column will contain one of the following values to describe the reason for the failure:
   - `PROVISIONING_REQUIRED`
   - `PENDING_UPGRADE`
   - `PROVISIONING_NOT_REQUIRED`
   - `PROVISIONING_FAILURE`

   See Also: "Understanding User Provisioning Statuses" on page 12-10 for more information on user provisioning statuses

Users Cannot Log In After Account Creation

To resolve typical problems that prevent users from logging in after account creation:

1. Examine the user provisioning statuses to identify the applications in which the user was not successfully provisioned by following the instructions described in "Using Provisioning Status to Identify Problems" on page C-16.
2. Identify the application provisioning approach for applications in which the user was not successfully provisioned.
   - For user accounts created with the Oracle Internet Directory Provisioning Console, examine the following Oracle Delegated Administration Services log file:

```
$ORACLE_HOME/opmn/logs/OC4J~OC4J_SECURITY~default_island~1
```

- For user accounts created with the PL/SQL plug-in or the Data Access Java plug-in, examine the following trace/audit file:

```
$ORACLE_HOME/ldap/odi/log/applicationType_realmName_E.trc
```

Monitoring Provisioning Execution Status with the Oracle Enterprise Manager 10*g* Application Server Control Console

You can use the Oracle Enterprise Manager 10*g* Application Server Control Console to monitor the provisioning execution status of provisioning integration profiles.

1. On the main Application Server Control Console page, select the name of the Oracle Application Server instance you want to manage in the **Standalone Instances** section. The Oracle Application Server home page opens for the selected instance.
2. In the **System Components** table, select **OID** in the Name column. The Oracle Internet Directory page opens. The status should be green if the required packages are installed properly. This does not indicate whether the Oracle directory integration and provisioning server is running or not.
3. To check the status of the servers, select **Directory Integration** to display the Directory Integration Platform Status page. This page displays the various running instances of Oracle directory integration and provisioning servers—including those for both provisioning and synchronization. The main data displayed for provisioning integration profiles in this window are:
   - Name of the subscribed application
   - Name of the organization for which the subscription was made
   - Status of the profile (ENABLED or DISABLED)
   - Change key in Oracle Internet Directory up to which the events have been propagated to the application that is represented by the profile
   - Last Execution Time
   - Last Successful Execution Time of the profile.
   - Errors, if any

> **Note:** The Directory Integration Platform Status page does not currently display the various event subscriptions for this profile

You can also get detailed output on provisioning integration status by running the `oidprovtool` utility with the operation argument `status`. The `oidprovtool` utility is located in the *$ORACLE_HOME*/bin directory.

> **Note:** The chapter on Oracle Directory Integration and Provisioning tools in the *Oracle Identity Management User Reference* for information on how to use the `oidprovtool` utility

Checklist for Debugging Provisioning

When troubleshooting provisioning, use the following as a checklist.

- On UNIX, use the following command to verify that the Oracle directory integration and provisioning server process (`odisrv`) is running:

  ```
  ps -ef | grep odisrv
  ```

  For Windows operating systems, obtain the value of process ID (PID) for the `odisrv` process from *$ORACLE_HOME*/ldap/log/oidmon.log. Then, launch Task Manager and click the Processes tab to verify that the process is running.

- Check whether there is also a directory integration and provisioning server instance running.

  If OracleAS Portal, Oracle Collaboration Suite, or another component needs provisioning, then there is probably a directory integration and provisioning server provisioning process running as instance 1 on configuration set 0. In this case, you should start your directory integration and provisioning server as instance 2 with either the default `configset=1` argument or using your custom created configuration set number.

  Check *$ORACLE_HOME*/ldap/log/odisrv0*x*.log. When the provisioning integration service is running, it logs to odisrv01.log. The directory synchronization service then logs to odisrv02.log.

- Verify that the profile is enabled by using the Oracle Directory Integration and Provisioning Server Administration tool or DIP Tester.

- Verify that trace files are being generated. The trace file can be found at: *$ORACLE_HOME*/ldap/odi/log/*profilename*.trc

  If no trace file is generated, then check the odisrv0*x*.log for possible problems in startup of the directory integration and provisioning server, as described earlier in this list.

- Verify that correct syntax is used to start the directory integration and provisioning server. For example:

  ```
  oidctl connect=asdb server=odisrv instance=2 configset=1 flags="host=myhost
  port=3060" start
  ```

- For debugging, verify that the value of the debug flag set to 63 when starting the directory integration and provisioning server, as follows:

  ```
  oidctl connect=asdb server=odisrv instance=2 configset=1 flags="host=myhost
  port=3060 debug=63" start
  ```

- Edit the profile and set the debug level to 63 by using the Oracle Directory Integration and Provisioning Server Administration tool or DIP Tester.

- Validate the all required parameters in the profile.

  See Also:

  Oracle MetaLink Note: 261342.1—Understanding DIP Mapping Files available on Oracle MetaLink at `http://metalink.oracle.com/`

  "Configuring Mapping Rules" on page 6-3

- Verify that you are using the Oracle Internet Directory 10*g* (10.1.2) version of the Oracle Directory Integration and Provisioning Server Administration tool or Oracle Directory Manager to update the profile. Previous releases of these utilities display different information on the Profile tab pages and should not be used.

- If you are using the PL/SQL plug-in, use `sqlplus` to verify that you can connect to the provisioning-integrated application.

  See Also: MetaLink Note: 265397.1—Password Policy Expires available on Oracle MetaLink at `http://metalink.oracle.com/`

Troubleshooting Synchronization

This section describes how to troubleshoot synchronization with Oracle Directory Integration and Provisioning. It contains these topics:

- Oracle Directory Integration and Provisioning Server Synchronization Process Flow
- Checklist for Debugging Synchronization
- Sample Valid Trace Files in Debug Level 63 Mode

Oracle Directory Integration and Provisioning Server Synchronization Process Flow

When debugging synchronization issues between Oracle Internet Directory and a connected directory, it helps to understand the synchronization process flow of the Oracle directory integration and provisioning server.

Oracle Directory Integration and Provisioning Server Synchronization Process Flow for an Import Profile

The Oracle directory integration and provisioning server reads all import profiles at startup. For each profile that is set to `ENABLE`, the Oracle directory integration and provisioning server performs the following tasks during the synchronization process:

1. Connects to a third-party directory
2. Gets the value of the last change key from the connected directory
3. Connects to Oracle Internet Directory
4. Gets the value of the profile's last applied change key from Oracle Internet Directory
5. For SunONE connections, the Oracle directory integration and provisioning server searches the remote change logs for entries greater than the value of the last applied change key and less than or equal to the value of the last change key. For Active Directory connections, the Oracle directory integration and provisioning server searches for this information in the remote directory's `uSNChanged` values. For other types of connectors, such as the Oracle Human Resources connector, the Oracle directory integration and provisioning server performs similar types of searches, although the method by which data is exchanged varies according to the type of connection.
6. Maps the data values from the connected directory to Oracle Internet Directory values
7. Creates an Oracle Internet Directory change record
8. Processes change (add, change, delete)
9. Updates the Oracle Internet Directory import profile with the last execution times and the last applied change key from the connected directory
10. Enters sleep mode for the number of seconds specified for the synchronization interval

Oracle Directory Integration and Provisioning Server Synchronization Process Flow for an Export Profile

The Oracle directory integration and provisioning server reads all export profiles at startup. For each profile that is set to `ENABLE`, the Oracle directory integration and provisioning server performs the following tasks during the synchronization process:

1. Connects to a third-party directory
2. Connects to Oracle Internet Directory
3. Gets the value for the last change key from Oracle Internet Directory
4. Gets the value of the profile's last applied change key from Oracle Internet Directory
5. For SunONE connections, the Oracle directory integration and provisioning server searches the remote change logs for entries greater than the value of the last applied change key and less than or equal to the value of the last change key. For Active Directory connections, the Oracle directory integration and provisioning server searches for this information in the remote directory's `uSNChanged` values. For other types of connectors, such as the Oracle Human Resources connector, the Oracle directory integration and provisioning server performs similar types of searches, although the method by which data is exchanged varies according to the type of connection.
6. Maps the data values from Oracle Internet Directory to the connected directory values
7. Creates a change record
8. Processes change (add, change, delete) on the connected directory
9. Updates the Oracle Internet Directory export profile with the last execution times and the last applied change key from Oracle Internet Directory
10. Enters sleep mode for the number of seconds specified for the synchronization interval

Checklist for Debugging Synchronization

When troubleshooting synchronization, use the following as a checklist.

- On UNIX, use the following command to verify that the Oracle directory integration and provisioning server process (`odisrv`) is running:

  ```
  ps -ef | grep odisrv
  ```

  For Windows operating systems, obtain the value of process ID (PID) for the `odisrv` process from *$ORACLE_HOME*/ldap/log/oidmon.log. Then, launch Task Manager and click the Processes tab to verify that the process is running.

- Check whether there is also a directory integration and provisioning server instance running.

  If OracleAS Portal, Oracle Collaboration Suite, or another component needs provisioning, then there is probably a directory integration and provisioning server provisioning process running as instance 1 on configuration set 0. In this case, you should start your directory integration and provisioning server as instance 2 with either the default `configset=1` argument or using your custom created configuration set number.

Check *$ORACLE_HOME*/ldap/log/odisrv0*x*.log. When the provisioning integration service is running, it logs to odisrv01.log. The directory synchronization service then logs to odisrv02.log.

- Verify that the profile is enabled by using the Oracle Directory Integration and Provisioning Server Administration tool or DIP Tester.
- Verify that trace files are being generated. The trace file can be found at: *$ORACLE_HOME*/ldap/odi/log/*profilename*.trc

  If no trace file is generated, then check the odisrv0*x*.log for possible problems in startup of the directory integration and provisioning server, as described earlier in this list.
- Verify that correct syntax is used to start the directory integration and provisioning server. For example:

  ```
  oidctl connect=asdb server=odisrv instance=2 configset=1 flags="host=myhost
  port=3060" start
  ```

- For debugging, verify that the value of the debug flag set to 63 when starting the directory integration and provisioning server, as follows:

  ```
  oidctl connect=asdb server=odisrv instance=2 configset=1 flags="host=myhost
  port=3060 debug=63" start
  ```

- Edit the profile and set the debug level to 63 by using the Oracle Directory Integration and Provisioning Server Administration tool or DIP Tester.
- Validate the all required parameters in the profile.

  See Also:

  Oracle MetaLink Note: 261342.1—Understanding DIP Mapping Files available on Oracle MetaLink at `http://metalink.oracle.com/`

  "Configuring Mapping Rules" on page 6-3

- Verify that you are using the Oracle Internet Directory 10*g* (10.1.2) version of the Oracle Directory Integration and Provisioning Server Administration tool or Oracle Directory Manager to update the profile. Previous releases of these utilities display different information on the Profile tab pages and should not be used.
- Verify that the third-party LDAP directory server is running by executing the following command:

  ```
  ldapbind -h ldap_host -p ldap_port -D account -w password
  ```

- If the directory integration and provisioning server does not start or if it starts and then fails, then check the following:
  - The instance number and configset being used
  - Whether the `flags="host=xxx port=xxxx"` parameter is used with `oidctl`
  - The odisrv0*x*.log to see whether:
    * Whether the connector successfully started
    * Whether the password expired

  To re-register the connector, enter the following command:

  ```
  odisrvreg -p port -D cn=orcladmin -w passwd -h host
  ```

See Also: MetaLink Note: 265397.1—Password Policy Expires available on Oracle MetaLink at http://metalink.oracle.com/

Sample Valid Trace Files in Debug Level 63 Mode

The following is the beginning and end portions of a valid sample trace file for an Active Directory connector synchronized addition operation:

```
--------------------------------------------------------------------------------
Trace Log Started at Tue Jun 08 11:22:25 EDT 2004
--------------------------------------------------------------------------------
Command exec succesful
LDAP URL : (activedir.oracle.com:389 administrator@oracle.com
LDAP Connection success
Applied ChangeNum : 28017Available chg num = 28019
Reader Initialised !!
LDAP URL : (sun1:3060 cn=odisrv+orclhostname=sun1,cn=odi,cn=oracle internet
directory
LDAP Connection success
Writer Initialised!!
MapEngine Initialised!!
Filter Initialised!!
searchF :
CHGLOGFILTER : (&(USNChanged>=28018)(USNChanged<=28022))
Search Time 8
Search Successful till # 28022
Search Changes Done
Changenumber uSNChanged: 28022
targetdn distinguishedName: CN=Test User56,CN=Users,DC=US,DC=ORACLE,DC=com
ChangeRecord : ----------
Changetype: 4
ChangeKey: CN=Test User56,CN=Users,DC=US,DC=ORACLE,DC=com
Attributes:
Class: null Name: ou Type: null ChgType: 1 Value: [ ]
Class: null Name: objectGUID Type: null ChgType: 2 Value: [[B@d0a5d9]

...

Class: null Name: mail Type: null ChgType: 1 Value: [ ]
Class: null Name: displayname Type: null ChgType: 2 Value: [Test User56]
Class: null Name: cn Type: null ChgType: 2 Value: [Test User56]
Class: null Name: sn Type: null ChgType: 2 Value: [Test User56]
Class: null Name: krbprincipalname Type: null ChgType: 1 Value: [@ ]
Class: null Name: uid Type: null ChgType: 1 Value: [ ]
Class: null Name: orcluserprincipalname Type: null ChgType: 1 Value: [ ]
Class: null Name: orclsamaccountname Type: null ChgType: 2 Value: [$Test User56]
-----------
DN : CN=Test User56,cn=users,dc=us,dc=oracle,dc=com
Normalized DN : CN=Test User56,cn=users,dc=us,dc=oracle,dc=com
Processing modifyRadd Operation ..
Entry Not Found. Converting to an ADD op..
Processing Insert Operation ..
Performing createEntry..
Entry Added Successfully : CN=Test User56,cn=users,dc=us,dc=oracle,dc=com
Updated Attributes
orclodipLastExecutionTime: 20040608112226
orclOdipSynchronizationStatus: Synchronization Successful
orclodipLastSuccessfulExecutionTime: 20040608112226
```

The following is the beginning and end portions of a valid sample trace file for an Active Directory connector synchronized deletion operation:

```
--------------------------------------------------------------------------------
 Trace Log Started at Wed Aug 18 09:10:05 EDT 2004
--------------------------------------------------------------------------------
Command exec succesful
LDAP URL : (sun1.mycompany.com:389 administrator@mycompany.com
LDAP Connection success
Applied ChangeNum : 31940Available chg num = 31940
Reader Initialised !!
LDAP URL : (sun2.mycompany.com:3060 cn=odisrv+orclhostname=sun2,cn=odi,cn=oracle
internet directory
LDAP Connection success
Writer Initialised!!
MapEngine Initialised!!
Filter Initialised!!
searchF :
CHGLOGFILTER : (&(USNChanged>=31941)(USNChanged<=31941))
Search Time 10
Search Successful till # 31941
Search Changes Done
Changenumber uSNChanged: 31941
Deleted isDeleted: TRUE
Deleted isDeleted: TRUE
ChangeRecord : ----------
Changetype: 1
ChangeKey: *
Attributes:
Class: null Name: objectGUID Type: null ChgType: 3 Value: [[B@ece65]

...

Output ChangeRecord ChangeRecord : ----------
Changetype: 1
ChangeKey: *
Attributes:
Class: null Name: objectclass Type: null ChgType: 3 Value: [organizationalunit,
orclcontainer, orcladuser, orcluserv2, orcladgroup]
Class: null Name: krbprincipalname Type: null ChgType: 3 Value: [@ ]
Class: null Name: orclsamaccountname Type: null ChgType: 3 Value: [$ ]
Class: null Name: orclobjectguid Type: null ChgType: 3 Value:
[2xR7Nas8UUKtzmPk0jpSFg==]
-----------
DN : *
Normalized DN : cn=TUser2007,cn=users,dc=us,dc=oracle,dc=com
Processing Delete Operation ..
Deleted entry Successfully : cn=TUser2007,cn=users,dc=us,dc=oracle,dc=com
Updated Attributes
orclodipLastExecutionTime: 20040818091005
orclOdipSynchronizationStatus: Synchronization Successful
orclodipLastSuccessfulExecutionTime: 20040818091005
```

The following is the beginning and end portions of a valid sample trace file for an Active Directory connector synchronized modify operation:

```
--------------------------------------------------------------------------------
 Trace Log Started at Wed Sep 29 09:40:18 EDT 2004
--------------------------------------------------------------------------------
Command exec succesful
LDAP URL : (server.mycompany.com:389 administrator@mycompany.com
```

```
LDAP Connection success
Applied ChangeNum : 35322Available chg num = 35322
Reader Initialised !!
LDAP URL : (sun2.mycompany.com:3060 cn=odisrv+orclhostname=sun2,cn=odi,cn=oracle
internet directory
LDAP Connection success
Writer Initialised!!
MapEngine Initialised!!
Filter Initialised!!
searchF :
CHGLOGFILTER : (&(USNCreated>=35323)(USNCreated<=35323))
Search Time 7
Search Successful till # 35323
Search Changes Done
searchF :
CHGLOGFILTER : (&(USNChanged>=35323)(USNChanged<=35323)(USNCreated<=35322))
Search Time 15
Search Successful till # 35323
Changenumber uSNChanged: 35323
targetdn distinguishedName: CN=Test User111,CN=Users,DC=US,DC=ORACLE,DC=com
ChangeRecord : ----------
Changetype: 4
ChangeKey: CN=Test User111,CN=Users,DC=US,DC=ORACLE,DC=com
Attributes:
Class: null Name: distinguishedname Type: null ChgType: 1 Value: [ ]
Class: null Name: samaccountname,userprincipalname Type: null ChgType: 1 Value: [
]
Class: null Name: userprincipalname Type: null ChgType: 1 Value: [ ]

...

 Output ChangeRecord ChangeRecord : ----------
Changetype: 4
ChangeKey: cn=TUser111,cn=users,dc=us,dc=oracle,dc=com
Attributes:
Class: null Name: objectclass Type: null ChgType: 3 Value: [orcluserv2,
orcladuser, inetorgperson, person]
Class: null Name: orclObjectSID Type: null ChgType: 2 Value:
[AQUAAAAAAAUVAAAAiqcyP8CFOF0VJa9HCAYAAA==]
Class: null Name: orclObjectGUID Type: null ChgType: 2 Value:
[6uEo05+F/0CHj4PTpPCchQ==]
Class: null Name: mail Type: null ChgType: 2 Value: [Tuser111@oracle.com]
Class: null Name: displayName Type: null ChgType: 2 Value: [Test User111]
Class: null Name: cn Type: null ChgType: 2 Value: [TUser111]
Class: null Name: sn Type: null ChgType: 2 Value: [TUser111]
Class: null Name: krbPrincipalName Type: null ChgType: 1 Value: [@ ]
Class: null Name: uid Type: null ChgType: 2 Value: [TUser111]
Class: null Name: orclUserPrincipalName Type: null ChgType: 1 Value: [ ]
Class: null Name: orclSAMAccountName Type: null ChgType: 2 Value: [$TUser111]
Class: null Name: orclDefaultProfileGroup Type: null ChgType: 1 Value: [ ]
-----------
DN : cn=TUser111,cn=users,dc=us,dc=oracle,dc=com
Normalized DN : cn=TUser111,cn=users,dc=us,dc=oracle,dc=com
Processing modifyRadd Operation ..
Entry found. Converting To a Modify Operation..
Proceeding with checkNReplace..
Performing checkNReplace..
Naming attribute: cn
Naming attribute value: orclDefaultProfileGroup
Naming attribute value: orclSAMAccountName
```

```
Naming attribute value: orclUserPrincipalName
Naming attribute value: uid
Naming attribute value: krbPrincipalName
Naming attribute value: sn
Naming attribute value: cn
Naming attribute value: displayName
Naming attribute value: mail
Adding Attribute in OID : mail
Naming attribute value: orclObjectGUID
Naming attribute value: orclObjectSID
Total # of Mod Items : 1
Modified Entry Successfully : cn=TUser111,cn=users,dc=us,dc=oracle,dc=com
Replacing Attribute orclodipLastSuccessfulExecutionTime in the Profile with value
: 20040929094018
Removed Existing attribute
RePopulated Attribute..
Updated Attributes
orclodipLastExecutionTime: 20040929094018
orclOdipSynchronizationStatus: Synchronization Successful
orclodipLastSuccessfulExecutionTime: 20040929094018
```

Troubleshooting Integration with Microsoft Active Directory

This section contains these topics:

- Debugging the Active Directory Connector
- Debugging Windows Native Authentication
- Troubleshooting the Microsoft Active Directory External Authentication Plug-in

Debugging the Active Directory Connector

You can debug the Active Directory connector by using the `oditest` and `DIP Tester` utilities described in "Troubleshooting Synchronization" on page C-19.

To troubleshoot the Active Directory connector:

- Run `oditest` and enter the profile name as the value of the directory synchronization profile argument
- Examine the *$ORACLE_HOME*/ldap/odi/log/AgentChgImp.trc and *$ORACLE_HOME*/ldap/odi/log/AgentChgImp.aud files in a text editor for pertinent information

If more than one profile is enabled, then DIP Tester can be run against each of them.

> **See Also:** MetaLink Note: 276481.1—Troubleshooting Oracle Directory Integration and Provisioning Synchronization Issues available on Oracle MetaLink at `http://metalink.oracle.com/`

Debugging Windows Native Authentication

Once you have configured Windows native authentication (see "Configuring Windows Native Authentication" on page 18-39), you can enable logging for this feature at run time. Open the opmn.xml file, located in *$ORACLE_HOME*/opmn/conf, and add the following parameter:

```
-Djazn.debug.log.enable = {true | false}
```

Assigning a value of `true` to the parameter enables debugging while assigning a value of `false` disables it.

The boldface text in the following example show where you should place the parameter in opmn.xml:

```
<process-type id="OC4J_SECURITY" module-id="OC4J">
  <environment>
    <variable id="DISPLAY" value="sun1.us.oracle.com:0.0"/>
    <variable id="LD_LIBRARY_PATH" value="/private/ora1012/OraHome1/lib"/>
  </environment>
  <module-data>
    <category id="start-parameters">
      <data id="java-options" value="-server -Djazn.debug.log.enable=true
      -Djava.security.policy=/private/ora1012/OraHome1/j2ee/OC4J_SECURITY/
      config/java2.policy -Djava.awt.headless=true -Xmx512m
      -Djava.awt.headless=true"/>
      <data id="oc4j-options" value="-properties"/>
    </category>
    <category id="stop-parameters">
      <data id="java-options" value="-Djava.security.policy=/private/ora1012/
      OraHome1/j2ee/OC4J_SECURITY/config/java2.policy  -Djava.awt.headless=true"/>
    </category>
```

The log is written to the file OC4J~OC4J_SECURITY~default_island~1, found at *$ORACLE_HOME*/opmn/logs.

> **See Also:** MetaLink Note: 283268.1—Troubleshooting Oracle Application Server Single Sign-On Windows Native Authentication available on Oracle MetaLink at `http://metalink.oracle.com/`

Troubleshooting the Microsoft Active Directory External Authentication Plug-in

If you are experiencing unknown errors, then you can enable plug-in debugging as explained in "Debugging the Windows NT External Authentication Plug-in" on page 19-4.

> **See Also:** MetaLink Note: 277382.1—How to Configure the Oracle Internet Directory External Authentication Plug-In for Authentication via Microsoft Active Directory available on Oracle MetaLink at `http://metalink.oracle.com/`

Troubleshooting Integration with the SunONE Connector

You can debug the SunONE connector by using the `oditest` and `DIP Tester` utilities described in "Troubleshooting Synchronization" on page C-19.

To troubleshoot the SunONE import connector:

- Run `oditest` and enter `IplanetImport` as the value of the directory synchronization profile argument
- Examine the *$ORACLE_HOME*/ldap/odi/log/IplanetImport.trc and *$ORACLE_HOME*/ldap/odi/log/IplanetImport.aud files in a text editor for pertinent information

To troubleshoot the SunONE export connector:

- Run `oditest` and enter `IplanetExport` as the value of the directory synchronization profile argument

- Examine the *$ORACLE_HOME*/ldap/odi/log/IplanetExport.trc and *$ORACLE_HOME*/ldap/odi/log/IplanetExport.aud files in a text editor for pertinent information

If more than one profile is enabled, then DIP Tester can be run against each of them.

Need More Help?

You can find more solutions on Oracle *MetaLink*, http://metalink.oracle.com. If you do not find a solution for your problem, log a service request.

See Also:

- *Oracle Application Server Release Notes*, available on the Oracle Technology Network: http://www.oracle.com/technology/documentation/index.html

Glossary

access control item (ACI)

An attribute that determines who has what type of access to what directory data. It contains a set of rules for structural access items, which pertain to entries, and content access items, which pertain to attributes. Access to both structural and content access items may be granted to one or more users or groups.

access control list (ACL)

The group of access directives that you define. The directives grant levels of access to specific data for specific clients, or groups of clients, or both.

access control policy point

An entry that contains security directives that apply downward to all entries at lower positions in the **directory information tree (DIT)**.

ACI

See **access control item (ACI)**.

ACL

See **access control list (ACL)**.

ACP

See **access control policy point**.

administrative area

A subtree on a directory server whose entries are under the control (schema, ACL, and collective attributes) of a single administrative authority.

advanced symmetric replication (ASR)

See **Oracle Database Advanced Replication**

anonymous authentication

The process by which the directory authenticates a user without requiring a user name and password combination. Each anonymous user then exercises the privileges specified for anonymous users.

API

See **application program interface**.

application program interface

Programs to access the services of a specified application. For example, LDAP-enabled clients access directory information through programmatic calls available in the LDAP API.

ASR

See **Oracle Database Advanced Replication**

attribute

An item of information that describes some aspect of an entry. An entry comprises a set of attributes, each of which belongs to an **object class**. Moreover, each attribute has both a *type*, which describes the kind of information in the attribute, and a *value*, which contains the actual data.

attribute configuration file

In an Oracle Directory Integration Platform environment, a file that specifies attributes of interest in a connected directory.

attribute type

The kind of information an attribute contains, for example, `jobTitle`.

attribute uniqueness

An Oracle Internet Directory feature that ensures that no two specified attributes have the same value. It enables applications synchronizing with the enterprise directory to use attributes as unique keys.

attribute value

The particular occurrence of information appearing in that entry. For example, the value for the `jobTitle` attribute could be `manager`.

authentication

The process of verifying the identity of a user, device, or other entity in a computer system, often as a prerequisite to allowing access to resources in a system.

authorization

Permission given to a user, program, or process to access an object or set of objects.

binding

The process of authenticating to a directory.

central directory

In an Oracle Directory Integration Platform environment, the directory that acts as the central repository. In an Oracle Directory Integration and Provisioning environment, Oracle Internet Directory is the central directory.

certificate

An ITU x.509 v3 standard data structure that securely binds an identity to a public key. A certificate is created when an entity's public key is signed by a trusted identity: a **certificate authority (CA)**. This certificate ensures that the entity's information is correct and that the public key actually belongs to that entity.

certificate authority (CA)

A trusted third party that certifies that other entities—users, databases, administrators, clients, servers—are who they say they are. The certificate authority verifies the user's identity and grants a certificate, signing it with the certificate authority's private key.

certificate chain

An ordered list of certificates containing an end-user or subscriber certificate and its certificate authority certificates.

change logs

A database that records changes made to a directory server.

cipher suite

In SSL, a set of authentication, encryption, and data integrity algorithms used for exchanging messages between network nodes. During an SSL handshake, the two nodes negotiate to see which cipher suite they will use when transmitting messages back and forth.

cluster

A collection of interconnected usable whole computers that is used as a single computing resource. Hardware clusters provide high availability and scalability.

cold backup

The procedure to add a new **DSA** node to an existing replicating system by using the database copy procedure.

concurrency

The ability to handle multiple requests simultaneously. Threads and processes are examples of concurrency mechanisms.

concurrent clients

The total number of clients that have established a session with Oracle Internet Directory.

concurrent operations

The number of operations that are being executed on the directory from all of the concurrent clients. Note that this is not necessarily the same as the concurrent clients, because some of the clients may be keeping their sessions idle.

configset

See **configuration set entry**.

configuration set entry

A directory entry holding the configuration parameters for a specific instance of the directory server. Multiple configuration set entries can be stored and referenced at runtime. The configuration set entries are maintained in the subtree specified by the subConfigsubEntry attribute of the DSE, which itself resides in the associated **directory information base (DIB)** against which the servers are started.

connect descriptor

A specially formatted description of the destination for a network connection. A connect descriptor contains destination service and network route information.

The destination service is indicated by using its service name for the Oracle Database or its Oracle System Identifier (SID) for Oracle release 8.0 or version 7 databases. The network route provides, at a minimum, the location of the listener through use of a network address.

connected directory

In an Oracle Directory Integration Platform environment, an information repository requiring full synchronization of data between Oracle Internet Directory and itself—for example, an Oracle human Resources database.

consumer

A directory server that is the destination of replication updates. Sometimes called a slave.

contention

Competition for resources.

context prefix

The **DN** of the root of a **naming context**.

cryptography

The practice of encoding and decoding data, resulting in secure messages.

data integrity

The guarantee that the contents of the message received were not altered from the contents of the original message sent.

decryption

The process of converting the contents of an encrypted message (ciphertext) back into its original readable format (plaintext).

default knowledge reference

A **knowledge reference** that is returned when the base object is not in the directory, and the operation is performed in a naming context not held locally by the server. A default knowledge reference typically sends the user to a server that has more knowledge about the directory partitioning arrangement.

default identity management realm

In a hosted environment, one enterprise—for example, an application service provider—makes Oracle components available to multiple other enterprises and stores information for them. In such hosted environments, the enterprise performing the hosting is called the default identity management realm, and the enterprises that are hosted are each associated with their own identity management realm in the DIT.

default realm location

An attribute in the root Oracle Context that identifies the root of the default identity management realm.

delegated administrator

In a hosted environment, one enterprise—for example, an application service provider—makes Oracle components available to multiple other enterprises and stores information for them. In such an environment, a global administrator performs activities that span the entire directory. Other administrators—called delegated

administrators—may exercise roles in specific identity management realms, or for specific applications.

DES

Data Encryption Standard, a block cipher developed by IBM and the U.S. government in the 1970's as an official standard.

DIB

See **directory information base (DIB)**.

directory information base (DIB)

The complete set of all information held in the directory. The DIB consists of entries that are related to each other hierarchically in a **directory information tree (DIT)**.

directory information tree (DIT)

A hierarchical tree-like structure consisting of the DNs of the entries.

directory integration profile

In an Oracle Directory Integration Platform environment, an entry in Oracle Internet Directory that describes how Oracle Directory Integration and Provisioning communicates with external systems and what is communicated.

directory integration and provisioning server

In an Oracle Directory Integration Platform environment, the server that drives the synchronization of data between Oracle Internet Directory and a **connected directory**.

directory naming context

See **naming context**.

directory provisioning profile

A special kind of **directory integration profile** that describes the nature of provisioning-related notifications that Oracle Directory Integration and Provisioning sends to the directory-enabled applications

directory replication group (DRG)

The directory servers participating in a replication agreement.

directory server instance

A discrete invocation of a directory server. Different invocations of a directory server, each started with the same or different configuration set entries and startup flags, are said to be different directory server instances.

directory-specific entry (DSE)

An entry specific to a directory server. Different directory servers may hold the same DIT name, but have different contents—that is, the contents can be specific to the directory holding it. A DSE is an entry with contents specific to the directory server holding it.

directory synchronization profile

A special kind of **directory integration profile** that describes how synchronization is carried out between Oracle Internet Directory and an external system.

directory system agent (DSA)

The X.500 term for a directory server.

distinguished name (DN)

The unique name of a directory entry. It comprises all of the individual names of the parent entries back to the root.

DIS

See **directory integration and provisioning server**

DIT

See **directory information tree (DIT)**

DN

See **distinguished name (DN)**

DRG

See **directory replication group (DRG)**

DSA

See **directory system agent (DSA)**

DSE

See **directory-specific entry (DSE)**

DSA-specific entries. Different DSAs may hold the same DIT name, but have different contents. That is, the contents can be specific to the DSA holding it. A DSE is an entry with contents specific to the DSA holding it.

encryption

The process of disguising the contents of a message and rendering it unreadable (ciphertext) to anyone but the intended recipient.

entry

The building block of a directory, it contains information about an object of interest to directory users.

export agent

In an Oracle Directory Integration Platform environment, an agent that exports data out of Oracle Internet Directory.

export data file

In an Oracle Directory Integration Platform environment, the file that contains data exported by an **export agent**.

export file

See **export data file**.

external agent

A directory integration agent that is independent of Oracle directory integration and provisioning server. The Oracle directory integration and provisioning server does not provide scheduling, mapping, or error handling services for it. An external agent is

typically used when a third party metadirectory solution is integrated with the Oracle Directory Integration Platform.

failover

The process of failure recognition and recovery. In an Oracle Application Server Cold Failover Cluster (Infrastructure), an application running on one cluster node is transparently migrated to another cluster node. During this migration, clients accessing the service on the cluster see a momentary outage and may need to reconnect once the failover is complete.

fan-out replication

Also called a point-to-point replication, a type of replication in which a supplier replicates directly to a consumer. That consumer can then replicate to one or more other consumers. The replication can be either full or partial.

filter

A method of qualifying data, usually data that you are seeking. Filters are always expressed as DNs, for example: `cn=susie smith,o=acme,c=us`.

global administrator

In a hosted environment, one enterprise—for example, an application service provider—makes Oracle components available to multiple other enterprises and stores information for them. In such an environment, a global administrator performs activities that span the entire directory.

global unique identifier (GUID)

An identifier generated by the system and inserted into an entry when the entry is added to the directory. In a multimaster replicated environment, the GUID, not the DN, uniquely identifies an entry. The GUID of an entry cannot be modified by a user.

grace login

A login occurring within the specified period before password expiration.

group search base

In the Oracle Internet Directory default DIT, the node in the identity management realm under which all the groups can be found.

guest user

One who is not an anonymous user, and, at the same time, does not have a specific user entry.

GUID

See **global unique identifier (GUID)**.

handshake

A protocol two computers use to initiate a communication session.

hash

A number generated from a string of text with an algorithm. The hash value is substantially smaller than the text itself. Hash numbers are used for security and for faster access to data.

identity management

The process by which the complete security life cycle for network entities is managed in an organization. It typically refers to the management of an organization's application users, where steps in the security life cycle include account creation, suspension, privilege modification, and account deletion. The network entities managed may also include devices, processes, applications, or anything else that needs to interact in a networked environment. Entities managed by an identity management process may also include users outside of the organization, for example customers, trading partners, or Web services.

identity management realm

A collection of identities, all of which are governed by the same administrative policies. In an enterprise, all employees having access to the intranet may belong to one realm, while all external users who access the public applications of the enterprise may belong to another realm. An identity management realm is represented in the directory by a specific entry with a special object class associated with it.

identity management realm-specific Oracle Context

An Oracle Context contained in each identity management realm. It stores the following information:

- User naming policy of the identity management realm—that is, how users are named and located
- Mandatory authentication attributes
- Location of groups in the identity management realm
- Privilege assignments for the identity management realm—for example: who has privileges to add more users to the Realm.
- Application specific data for that Realm including authorizations

import agent

In an Oracle Directory Integration Platform environment, an agent that imports data into Oracle Internet Directory.

import data file

In an Oracle Directory Integration Platform environment, the file containing the data imported by an **import agent**.

inherit

When an object class has been derived from another class, it also derives, or inherits, many of the characteristics of that other class. Similarly, an attribute subtype inherits the characteristics of its supertype.

instance

See **directory server instance**.

integrity

The guarantee that the contents of the message received were not altered from the contents of the original message sent.

Internet Engineering Task Force (IETF)

The principal body engaged in the development of new Internet standard specifications. It is an international community of network designers, operators,

vendors, and researchers concerned with the evolution of the Internet architecture and the smooth operation of the Internet.

Internet Message Access Protocol (IMAP)

A protocol allowing a client to access and manipulate electronic mail messages on a server. It permits manipulation of remote message folders, also called mailboxes, in a way that is functionally equivalent to local mailboxes.

key

A string of bits used widely in cryptography, allowing people to encrypt and decrypt data; a key can be used to perform other mathematical operations as well. Given a cipher, a key determines the mapping of the plaintext to the ciphertext.

key pair

A **public key** and its associated **private key**.

See **public/private key pair**.

knowledge reference

The access information (name and address) for a remote **DSA** and the name of the **DIT** subtree that the remote DSA holds. Knowledge references are also called referrals.

latency

The time a client has to wait for a given directory operation to complete. Latency can be defined as wasted time. In networking discussions, latency is defined as the travel time of a packet from source to destination.

LDAP

See **Lightweight Directory Access Protocol (LDAP)**.

LDIF

See **LDAP Data Interchange Format (LDIF)**.

Lightweight Directory Access Protocol (LDAP)

A standard, extensible directory access protocol. It is a common language that LDAP clients and servers use to communicate. The framework of design conventions supporting industry-standard directory products, such as the Oracle Internet Directory.

LDAP Data Interchange Format (LDIF)

The set of standards for formatting an input file for any of the LDAP command-line utilities.

logical host

In an Oracle Application Server Cold Failover Cluster (Infrastructure), one or more disk groups and pairs of host names and IP addresses. It is mapped to a physical host in the cluster. This physical host impersonates the host name and IP address of the logical host

man-in-the-middle

A security attack characterized by the third-party, surreptitious interception of a message. The third-party, the *man-in-the-middle*, decrypts the message, re-encrypts it (with or without alteration of the original message), and retransmits it to the

originally-intended recipient—all without the knowledge of the legitimate sender and receiver. This type of security attack works only in the absence of **authentication**.

mapping rules file

In an Oracle Directory Integration Platform environment, the file that specifies mappings between Oracle Internet Directory attributes and those in a **connected directory**.

master definition site (MDS)

In replication, a master definition site is the Oracle Internet Directory database from which the administrator runs the configuration scripts.

master site

In replication, a master site is any site other than the master definition site that participates in LDAP replication.

matching rule

In a search or compare operation, determines equality between the attribute value sought and the attribute value stored. For example, matching rules associated with the `telephoneNumber` attribute could cause "(650) 123-4567" to be matched with either "(650) 123-4567" or "6501234567" or both. When you create an attribute, you associate a matching rule with it.

MD4

A one-way hash function that produces a 128-bit hash, or message digest. If as little as a single bit value in the file is modified, the MD4 checksum for the file will change. Forgery of a file in a way that will cause MD4 to generate the same result as that for the original file is considered extremely difficult.

MD5

An improved version of MD4.

MDS

See **master definition site (MDS)**

metadirectory

A directory solution that shares information between all enterprise directories, integrating them into one virtual directory. It centralizes administration, thereby reducing administrative costs. It synchronizes data between directories, thereby ensuring that it is consistent and up-to-date across the enterprise.

MTS

See **shared server**

multimaster replication

Also called peer-to-peer or *n*-way replication, a type of replication that enables multiple sites, acting as equals, to manage groups of replicated data. In a multimaster replication environment, each node is both a supplier and a consumer node, and the entire directory is replicated on each node.

naming attribute

The attribute used to compose the RDN of a new user entry created through Oracle Delegated Administration Services or Oracle Internet Directory Java APIs. The default value for this is `cn`.

naming context

A subtree that resides entirely on one server. It must be contiguous, that is, it must begin at an entry that serves as the top of the subtree, and extend downward to either leaf entries or **knowledge reference**s (also called referrals) to subordinate naming contexts. It can range in size from a single entry to the entire DIT.

native agent

In an Oracle Directory Integration Platform environment, an agent that runs under the control of the **directory integration and provisioning server**. It is in contrast to an **external agent**.

net service name

A simple name for a service that resolves to a connect descriptor. Users initiate a connect request by passing a user name and password along with a net service name in a connect string for the service to which they wish to connect:

```
CONNECT username/password@net_service_name
```

Depending on your needs, net service names can be stored in a variety of places, including:

- Local configuration file, `tnsnames.ora`, on each client
- Directory server
- Oracle Names server
- External naming service, such as NDS, NIS or CDS

nickname attribute

The attribute used to uniquely identify a user in the entire directory. The default value for this is `uid`. Applications use this to resolve a simple user name to the complete distinguished name. The user nickname attribute cannot be multi-valued—that is, a given user cannot have multiple nicknames stored under the same attribute name.

object class

A named group of attributes. When you want to assign attributes to an entry, you do so by assigning to that entry the object classes that hold those attributes.

All objects associated with the same object class share the same attributes.

OEM

See **Oracle Enterprise Manager**.

OID Control Utility

A command-line tool for issuing run-server and stop-server commands. The commands are interpreted and executed by the **OID Monitor** process.

OID Database Password Utility

The utility used to change the password with which Oracle Internet Directory connects to an Oracle database.

OID Monitor

The Oracle Internet Directory component that initiates, monitors, and terminates the Oracle directory server processes. It also controls the replication server if one is installed, and Oracle directory integration and provisioning server.

one-way function

A function that is easy to compute in one direction but quite difficult to reverse compute, that is, to compute in the opposite direction.

one-way hash function

A **one-way function** that takes a variable sized input and creates a fixed size output.

Oracle Call Interface (OCI)

An application programming interface (API) that enables you to create applications that use the native procedures or function calls of a third-generation language to access an Oracle database server and control all phases of SQL statement execution.

Oracle Delegated Administration Services

A set of individual, pre-defined services—called Oracle Delegated Administration Services units—for performing directory operations on behalf of a user. Oracle Internet Directory Self-Service Console makes it easier to develop and deploy administration solutions for both Oracle and third-party applications that use Oracle Internet Directory.

Oracle Directory Integration Platform

A component of **Oracle Internet Directory**. It is a framework developed to integrate applications around a central LDAP directory like Oracle Internet Directory.

Oracle directory integration and provisioning server

In an Oracle Directory Integration Platform environment, a daemon process that monitors Oracle Internet Directory for change events and takes action based on the information present in the **directory integration profile**.

Oracle Directory Manager

A Java-based tool with a graphical user interface for administering Oracle Internet Directory.

Oracle Enterprise Manager

A separate Oracle product that combines a graphical console, agents, common services, and tools to provide an integrated and comprehensive systems management platform for managing Oracle products.

Oracle Identity Management

An infrastructure enabling deployments to manage centrally and securely all enterprise identities and their access to various applications in the enterprise.

Oracle Internet Directory

A general purpose directory service that enables retrieval 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 the Oracle Database.

Oracle Net Services

The foundation of the Oracle family of networking products, allowing services and their client applications to reside on different computers and communicate. The main function of Oracle Net Services is to establish network sessions and transfer data between a client application and a server. Oracle Net Services is located on each computer in the network. Once a network session is established, Oracle Net Services acts as a data courier for the client and the server.

Oracle PKI certificate usages

Defines Oracle application types that a **certificate** supports.

Oracle Wallet Manager

A Java-based application that security administrators use to manage public-key security credentials on clients and servers.

See Also: *Oracle Advanced Security Administrator's Guide*

Oracle Database Advanced Replication

A feature in the Oracle Database that enables database tables to be kept synchronized across two Oracle databases.

other information repository

In an Oracle Directory Integration and Provisioning environment, in which Oracle Internet Directory serves as the **central directory**, any information repository except Oracle Internet Directory.

partition

A unique, non-overlapping directory naming context that is stored on one directory server.

peer-to-peer replication

Also called multimaster replication or *n*-way replication. A type of replication that enables multiple sites, acting as equals, to manage groups of replicated data. In such a replication environment, each node is both a supplier and a consumer node, and the entire directory is replicated on each node.

PKCS #12

A **public-key encryption** standard (PKCS). RSA Data Security, Inc. PKCS #12 is an industry standard for storing and transferring personal authentication credentials—typically in a format called a **wallet**.

plaintext

Message text that has not been encrypted.

point-to-point replication

Also called fan-out replication is a type of replication in which a supplier replicates directly to a consumer. That consumer can then replicate to one or more other consumers. The replication can be either full or partial.

primary node

In an Oracle Application Server Cold Failover Cluster (Infrastructure), the cluster node on which the application runs at any given time.

See Also: **secondary node** on page Glossary-16

private key

In public-key cryptography, this key is the secret key. It is primarily used for decryption but is also used for encryption with digital signatures.

provisioning agent

An application or process that translates Oracle-specific provisioning events to external or third-party application-specific events.

provisioned applications

Applications in an environment where user and group information is centralized in Oracle Internet Directory. These applications are typically interested in changes to that information in Oracle Internet Directory.

profile

See **directory integration profile**

proxy user

A kind of user typically employed in an environment with a middle tier such as a firewall. In such an environment, the end user authenticates to the middle tier. The middle tier then logs into the directory on the end user's behalf. A proxy user has the privilege to switch identities and, once it has logged into the directory, switches to the end user's identity. It then performs operations on the end user's behalf, using the authorization appropriate to that particular end user.

public key

In public-key cryptography this key is made public to all, it is primarily used for encryption but can be used for verifying signatures.

public-key cryptography

Cryptography based on methods involving a public key and a private key.

public-key encryption

The process in which the sender of a message encrypts the message with the public key of the recipient. Upon delivery, the message is decrypted by the recipient using the recipient's private key.

public/private key pair

A mathematically related set of two numbers where one is called the private key and the other is called the public key. Public keys are typically made widely available, while private keys are available only to their owners. Data encrypted with a public key can only be decrypted with its associated private key and vice versa. Data encrypted with a public key cannot be decrypted with the same public key.

realm search base

An attribute in the root Oracle Context that identifies the entry in the DIT that contains all identity management realms. This attribute is used when mapping a simple realm name to the corresponding entry in the directory.

referral

Information that a directory server provides to a client and which points to other servers the client must contact to find the information it is requesting.

See also **knowledge reference**.

relational database

A structured collection of data that stores data in tables consisting of one or more rows, each containing the same set of columns. Oracle makes it very easy to link the data in multiple tables. This is what makes Oracle a relational database management system, or RDBMS. It stores data in two or more tables and enables you to define relationships between the tables. The link is based on one or more fields common to both tables.

replica

Each copy of a naming context that is contained within a single server.

RDN

See **relative distinguished name (RDN)**.

registry entry

An entry containing runtime information associated with invocations of Oracle directory servers, called a **directory server instance**. Registry entries are stored in the directory itself, and remain there until the corresponding directory server instance stops.

relative distinguished name (RDN)

The local, most granular level entry name. It has no other qualifying entry names that would serve to uniquely address the entry. In the example, `cn=Smith,o=acme,c=US`, the RDN is `cn=Smith`.

remote master site (RMS)

In a replicated environment, any site, other than the **master definition site (MDS)**, that participates in Oracle Database Advanced Replication.

replication agreement

A special directory entry that represents the replication relationship among the directory servers in a **directory replication group (DRG)**.

response time

The time between the submission of a request and the completion of the response.

root DSE

See **root directory specific entry**.

root directory specific entry

An entry storing operational information about the directory. The information is stored in a number of attributes.

Root Oracle Context

In the Oracle Identity Management infrastructure, the Root Oracle Context is an entry in Oracle Internet Directory containing a pointer to the default identity management realm in the infrastructure. It also contains information on how to locate an identity management realm given a simple name of the realm.

SASL

See **Simple Authentication and Security Layer (SASL)**

scalability

The ability of a system to provide throughput in proportion to, and limited only by, available hardware resources.

schema

The collection of attributes, object classes, and their corresponding matching rules.

secondary node

In an Oracle Application Server Cold Failover Cluster (Infrastructure), the cluster node to which an application is moved during a failover.

See Also: **primary node** on page Glossary-13

Secure Hash Algorithm (SHA)

An algorithm that takes a message of less than 264 bits in length and produces a 160-bit message digest. The algorithm is slightly slower than MD5, but the larger message digest makes it more secure against brute-force collision and inversion attacks.

Secure Socket Layer (SSL)

An industry standard protocol designed by Netscape Communications Corporation for securing network connections. SSL provides authentication, encryption, and data integrity using public key infrastructure (PKI).

service time

The time between the initiation of a request and the completion of the response to the request.

session key

A key for symmetric-key cryptosystems that is used for the duration of one message or communication session.

SGA

See **System Global Area (SGA)**.

SHA

See **Secure Hash Algorithm (SHA)**.

shared server

A server that is configured to allow many user processes to share very few server processes, so the number of users that can be supported is increased. With shared server configuration, many user processes connect to a dispatcher. The dispatcher directs multiple incoming network session requests to a common queue. An idle shared server process from a shared pool of server processes picks up a request from the queue. This means a small pool of server processes can server a large amount of clients. Contrast with dedicated server.

sibling

An entry that has the same parent as one or more other entries.

simple authentication

The process by which the client identifies itself to the server by means of a DN and a password which are not encrypted when sent over the network. In the simple

authentication option, the server verifies that the DN and password sent by the client match the DN and password stored in the directory.

Simple Authentication and Security Layer (SASL)

A method for adding authentication support to connection-based protocols. To use this specification, a protocol includes a command for identifying and authenticating a user to a server and for optionally negotiating a security layer for subsequent protocol interactions. The command has a required argument identifying a SASL mechanism.

single key-pair wallet

A **PKCS #12**-format **wallet** that contains a single user **certificate** and its associated **private key**. The **public key** is imbedded in the certificate.

slave

See **consumer**.

SLAPD

Standalone LDAP daemon.

smart knowledge reference

A **knowledge reference** that is returned when the knowledge reference entry is in the scope of the search. It points the user to the server that stores the requested information.

specific administrative area

Administrative areas control:

- Subschema administration
- Access control administration
- Collective attribute administration

A *specific* administrative area controls one of these aspects of administration. A specific administrative area is part of an autonomous administrative area.

sponsor node

In replication, the node that is used to provide initial data to a new node.

SSL

See **Secure Socket Layer (SSL)**.

subACLSubentry

A specific type of subentry that contains ACL information.

subclass

An object class derived from another object class. The object class from which it is derived is called its **superclass**.

subentry

A type of entry containing information applicable to a group of entries in a subtree. The information can be of these types:

- Access control policy points
- Schema rules

- Collective attributes

Subentries are located immediately below the root of an administrative area.

subordinate reference

A knowledge reference pointing downward in the DIT to a naming context that starts immediately below an entry.

subschema DN

The list of DIT areas having independent schema definitions.

subSchemaSubentry

A specific type of **subentry** containing schema information.

subtype

An attribute with one or more options, in contrast to that same attribute without the options. For example, a `commonName` (`cn`) attribute with American English as an option is a subtype of the `commonName` (`cn`) attribute without that option. Conversely, the `commonName` (`cn`) attribute without an option is the **supertype** of the same attribute with an option.

super user

A special directory administrator who typically has full access to directory information.

superclass

The object class from which another object class is derived. For example, the object class `person` is the superclass of the object class `organizationalPerson`. The latter, namely, `organizationalPerson`, is a **subclass** of `person` and inherits the attributes contained in `person`.

superior reference

A knowledge reference pointing upward to a DSA that holds a naming context higher in the DIT than all the naming contexts held by the referencing DSA.

supertype

An attribute without options, in contrast to the same attribute with one or more options. For example, the `commonName` (`cn`) attribute without an option is the supertype of the same attribute with an option. Conversely, a `commonName` (`cn`) attribute with American English as an option is a **subtype** of the `commonName` (`cn`) attribute without that option.

supplier

In replication, the server that holds the master copy of the naming context. It supplies updates from the master copy to the **consumer** server.

System Global Area (SGA)

A group of shared memory structures that contain data and control information for one Oracle database instance. If multiple users are concurrently connected to the same instance, the data in the instance SGA is shared among the users. Consequently, the SGA is sometimes referred to as the "shared global area." The combination of the background processes and memory buffers is called an Oracle instance.

system operational attribute

An attribute holding information that pertains to the operation of the directory itself. Some operational information is specified by the directory to control the server, for example, the time stamp for an entry. Other operational information, such as access information, is defined by administrators and is used by the directory program in its processing.

TLS

See **Transport Layer Security (TLS)**

think time

The time the user is not engaged in actual use of the processor.

throughput

The number of requests processed by Oracle Internet Directory for each unit of time. This is typically represented as "operations per second."

Transport Layer Security (TLS)

A protocol providing communications privacy over the Internet. The protocol enables client/server applications to communicate in a way that prevents eavesdropping, tampering, or message forgery.

trusted certificate

A third party identity that is qualified with a level of trust. The trust is used when an identity is being validated as the entity it claims to be. Typically, the certificate authorities you trust issue user certificates.

trustpoint

See **trusted certificate**.

UTF-16

16-bit encoding of **Unicode**.The Latin-1 characters are the first 256 code points in this standard.

Unicode

A type of universal character set, a collection of 64K characters encoded in a 16-bit space. It encodes nearly every character in just about every existing character set standard, covering most written scripts used in the world. It is owned and defined by Unicode Inc. Unicode is canonical encoding which means its value can be passed around in different locales. But it does not guarantee a round-trip conversion between it and every Oracle character set without information loss.

UNIX Crypt

The UNIX encryption algorithm.

user search base

In the Oracle Internet Directory default DIT, the node in the identity management realm under which all the users are placed.

UTC (Coordinated Universal Time)

The standard time common to every place in the world. Formerly and still widely called Greenwich Mean Time (GMT) and also World Time, UTC nominally reflects the

mean solar time along the Earth's prime meridian. UTC is indicated by a z at the end of the value, for example, 200011281010z.

UTF-8

A variable-width 8-bit encoding of **Unicode** that uses sequences of 1, 2, 3, or 4 bytes for each character. Characters from 0-127 (the 7-bit ASCII characters) are encoded with one byte, characters from 128-2047 require two bytes, characters from 2048-65535 require three bytes, and characters beyond 65535 require four bytes. The Oracle character set name for this is AL32UTF8 (for the Unicode 3.1 standard).

virtual host name

In an Oracle Application Server Cold Failover Cluster (Infrastructure), the host name corresponding to this virtual IP address.

virtual IP address

In an Oracle Application Server Cold Failover Cluster (Infrastructure), each physical node has its own physical IP address and physical host name. To present a single system image to the outside world, the cluster uses a dynamic IP address that can be moved to any physical node in the cluster. This is called the virtual IP address.

wallet

An abstraction used to store and manage security credentials for an individual entity. It implements the storage and retrieval of credentials for use with various cryptographic services. A wallet resource locator (WRL) provides all the necessary information to locate the wallet.

wait time

The time between the submission of the request and initiation of the response.

X.509

A popular format from ISO used to sign public keys.

Index

A

B

C

D

E

F

G

H

I

K

L

M

N

O

P

R

S

T

U

V

W