Contents

Preface ........................................................................................................................................................... xxxv
  Audience .................................................................................................................................................... xxxv
  Documentation Accessibility ................................................................................................................ xxxv
  Related Documents ................................................................................................................................. xxxvi
  Conventions ............................................................................................................................................. xxxvii

What’s New in Oracle Internet Directory? .............................................................................................. xxxix
  New Features Introduced with Oracle Internet Directory 10g (10.1.4.0.1) ........................................ xxxix
  New Features Introduced with Oracle Internet Directory 10g Release 2 (10.1.2) ................................ xliii
  New Features Introduced with Oracle Internet Directory 10g (9.0.4) ................................................ xliii
  About Oracle Internet Directory Release 9.2 ......................................................................................... xlvii
  New Features Introduced with Oracle Internet Directory Release 9.0.2 ........................................... xlviii
  New Features Introduced with Oracle Internet Directory Release 3.0.1 ............................................. li
  New Features Introduced with Oracle Internet Directory Release 2.1.1 ............................................. liii

Part I  Getting Started

1  Links to Common Tasks
  Object Classes and Attributes ............................................................................................................... 1-1
  Replication ........................................................................................................................................... 1-2
  Security, Password Policies, and User Accounts ............................................................................. 1-2
  Realms ................................................................................................................................................. 1-3
  Server Processes, Instances, and Configuration Set Entries ......................................................... 1-3
  System Operational Attributes ........................................................................................................ 1-5
  Naming Contexts ............................................................................................................................... 1-5
  Binds, Connections, Aliases, and Directory Discovery .................................................................... 1-5
  Referential Integrity .......................................................................................................................... 1-6
  Entries ............................................................................................................................................... 1-6
  Groups ............................................................................................................................................... 1-7
  Logging, Auditing, and Monitoring ................................................................................................... 1-8
  Tuning ................................................................................................................................................ 1-8
  Garbage Collection .......................................................................................................................... 1-9
  Server Chaining and Data Migration ............................................................................................... 1-9
  Plug-ins .............................................................................................................................................. 1-9
2 Introduction to LDAP and Oracle Internet Directory

What Is a Directory?................................................................................................................. 2-1
The Expanding Role of Online Directories........................................................................... 2-1
The Problem: Too Many Special-Purpose Directories ......................................................... 2-2
LDAP and Simplified Directory Management .................................................................... 2-3
LDAP Version 3..................................................................................................................... 2-3
Oracle Identity Management ............................................................................................... 2-4
What Is Oracle Internet Directory?..................................................................................... 2-5
Overview of Oracle Internet Directory ............................................................................... 2-5
Components of Oracle Internet Directory .......................................................................... 2-6
Advantages of Oracle Internet Directory ............................................................................ 2-7
Scalability .............................................................................................................................. 2-7
High Availability .................................................................................................................. 2-7
Security ................................................................................................................................. 2-7
Integration with the Oracle Environment ............................................................................ 2-7
How Oracle Components Use Oracle Internet Directory............................................... 2-7
Easier and More Cost-Effective Administration of Applications ....................................... 2-8
Tighter Security Through Centralized Security Policy Administration ............................... 2-8
Integration of Multiple Directories ....................................................................................... 2-9

3 Directory Concepts and Architecture

Oracle Internet Directory Architecture .................................................................................. 3-1
An Oracle Internet Directory Node ...................................................................................... 3-2
An Oracle Directory Server Instance .................................................................................... 3-3
Directory Metadata .............................................................................................................. 3-4
Configuration Set Entries ..................................................................................................... 3-6
Example: How Oracle Internet Directory Works................................................................. 3-6
Entries .................................................................................................................................. 3-7
Distinguished Names (DNs) and Directory Information Trees (DITs) ................................. 3-7
Entry Caching ....................................................................................................................... 3-8
Attributes ............................................................................................................................. 3-8
Kinds of Attribute Information ............................................................................................ 3-9
Single-Valued and Multivalued Attributes ......................................................................... 3-10
Common LDAP Attributes ............................................................................................... 3-10
Attribute Syntax .................................................................................................................. 3-11
Attribute Matching Rules ................................................................................................. 3-11
Attribute Options ............................................................................................................... 3-11
Object Classes ..................................................................................................................... 3-12
Subclasses, Superclasses, and Inheritance .......................................................................... 3-12
Object Class Types ............................................................................................................ 3-12
Structural Object Classes .................................................................................................. 3-13
Auxiliary Object Classes .................................................................................................... 3-13
Abstract Object Classes .................................................................................................... 3-13
Naming Contexts .................................................................................................................. 3-14
Security ............................................................................................................................... 3-14
Globalization Support ......................................................................................................... 3-15
Distributed Directories ........................................................................................................................... 3-16
  Directory Replication ............................................................................................................................ 3-17
  Directory Partitioning ......................................................................................................................... 3-18

Knowledge References and Referrals .................................................................................................. 3-19

Oracle Delegated Administration Services and the Oracle Internet Directory Self-Service Console ........................................................................................................................................ 3-21

The Service Registry and Service to Service Authentication ................................................................. 3-21

Oracle Directory Integration Platform .................................................................................................. 3-22

Oracle Internet Directory and Identity Management ............................................................................. 3-22
  About Identity Management ................................................................................................................ 3-23
  About the Oracle Identity Management Infrastructure ......................................................................... 3-23
  Identity Management Realms ................................................................................................................ 3-25
    Default Identity Management Realm .............................................................................................. 3-25
    Identity Management Policies ........................................................................................................... 3-25

Resource Information ............................................................................................................................ 3-26
  Resource Type Information .................................................................................................................. 3-26
  Resource Access Information .............................................................................................................. 3-26
  Location of Resource Information in the DIT ...................................................................................... 3-27

4 Post-Installation Tasks and Information

Task 1: Reset the Default Security Configuration .................................................................................. 4-1
Task 2: Reset the Default Password for the Database ........................................................................... 4-3
Task 3: Run the OID Database Statistics Collection Tool ....................................................................... 4-3
Tasks to Perform After Upgrading from Release 9.0.2 ....................................................................... 4-3
  Set ACL Policy on Groups Container after Upgrade from Release 9.0.2 .............................................. 4-3

Determining LDAP Port Assignment on UNIX and Linux .................................................................... 4-4

5 Directory Administration and Monitoring Tools

Using Oracle Identity Management Grid Control Plug-in ....................................................................... 5-1

Using Oracle Directory Manager .......................................................................................................... 5-1
  Starting Oracle Directory Manager ..................................................................................................... 5-2
  Connecting to a Directory Server by Using Oracle Directory Manager .............................................. 5-2
  Navigating Oracle Directory Manager ................................................................................................ 5-3
    Overview of Oracle Directory Manager ............................................................................................ 5-3
    The Oracle Directory Manager Menu Bar ........................................................................................ 5-3
    The Oracle Directory Manager Toolbar ........................................................................................... 5-5
  Connecting to Additional Directory Servers by Using Oracle Directory Manager ............................ 5-5
  Disconnecting from a Directory Server by Using Oracle Directory Manager ..................................... 5-6
  Configuring the Display and Duration of Searches in Oracle Directory Manager ............................... 5-6
  Performing Administrative Tasks by Using Oracle Directory Manager ............................................. 5-7

Using Oracle Internet Directory Server Manageability ......................................................................... 5-8

Using Command-Line Tools .................................................................................................................. 5-8
  Command-Line Tools for Starting, Stopping, and Monitoring Oracle Internet Directory Servers ......... 5-9
  Command-Line Tools for Managing Entries and Attributes ................................................................ 5-10
  Command-Line Tools for Performing Bulk Operations ....................................................................... 5-10
6 Process Control of Oracle Internet Directory Components

Tools and Daemons Important to Oracle Internet Directory Process Control .......................................................... 6-1
Oracle Internet Directory Integration with OPMN ............................................................................................ 6-2
Semantics of OPMN Monitoring Oracle Internet Directory .................................................................................. 6-2
Oracle Internet Directory Snippet in OPMN.XML ............................................................................................ 6-2
Semantics of OPMN Starting Oracle Internet Directory ................................................................................... 6-2
Semantics of OPMN Stopping Oracle Internet Directory ................................................................................... 6-3
Semantics of OPMN Monitizing OIDMON ......................................................................................................... 6-3
Oracle Internet Directory Process Control—Best Practices ........................................................................... 6-3
Changing the Configuration of the Default OID LDAP Server Instance ......................................................... 6-4
Configuring Additional Oracle Internet Directory LDAP Server Instances .................................................. 6-5
Deconfiguring the Default Oracle Internet Directory LDAP Server Instance ................................................. 6-5
Configuring an Instance of the Oracle Internet Directory Replication Server .............................................. 6-6
Configuring an Oracle Directory Integration Platform Server Instance ..................................................... 6-6
OIDMON and the ODS_PROCESS Table ........................................................................................................... 6-6
OIDCTL Process Control Semantics .................................................................................................................. 6-7

Part II Basic Directory Administration

7 Oracle Directory Server Administration

Managing Server Configuration Set Entries ........................................................................................................ 7-1
Preliminary Considerations for Managing Configuration Set Entries .............................................................. 7-2
Managing Server Configuration Set Entries by Using Oracle Directory Manager ........................................ 7-3
Viewing Configuration Set Entries by Using Oracle Directory Manager ......................................................... 7-3
Adding Configuration Set Entries by Using Oracle Directory Manager ......................................................... 7-4
Modifying Configuration Set Entries by Using Oracle Directory Manager ................................................... 7-5
Deleting Configuration Set Entries by Using Oracle Directory Manager ...................................................... 7-5
Managing Server Configuration Set Entries by Using Command-Line Tools ................................................ 7-5
Adding Configuration Set Entries by Using ldapadd .................................................................................... 7-6
Modifying and Deleting Configuration Set Entries by Using ldapmodify ................................................... 7-6
Setting System Operational Attributes .......................................................................................................... 7-7
Setting System Operational Attributes by Using Oracle Directory Manager .............................................. 7-7
Setting System Operational Attributes by Using ldapmodify ................................................................... 7-7
Managing Naming Contexts .............................................................................................................................. 7-8
Publishing Naming Contexts by Using Oracle Directory Manager ................................................................. 7-8
Publishing Naming Contexts by Using ldapmodify ....................................................................................... 7-8
Managing Super Users, Guest Users, and Proxy Users .................................................................................... 7-8
About Super Users, Guest, Users, and Proxy Users .......................................................................................... 7-9
Managing Super Users, Guest Users, and Proxy Users by Using Oracle Directory Manager ............................... 7-9
Managing Super Users, Guest Users, and Proxy Users by Using ldapmodify .................................................. 7-10
Managing Anonymous Binds .............................................................................................................................. 7-10
Viewing Active Server Instance Information ........................................................................................................ 7-11
Closing Idle LDAP Connections ............................................................................................................................. 7-11
Changing the Password to the Oracle Internet Directory Database Server .......................................................... 7-11
Dereferencing Alias Entries .................................................................................................................................. 7-12
About Alias Entries .................................................................................................................................................. 7-12
Examples: Using Alias Entry Dereferencing ........................................................................................................ 7-13
  Example: Adding an Alias Entry ........................................................................................................................ 7-13
  Examples: Searching the Directory with Alias Entries ....................................................................................... 7-14
  Example: Modifying Alias Entries ..................................................................................................................... 7-15
Success and Error Messages .................................................................................................................................. 7-15
Locating Directory Servers in a Distributed Environment ...................................................................................... 7-16
  Static Directory Server Discovery by Using the Directory Server Usage File (ldap.ora) .................................. 7-16
  Dynamic Directory Server Discovery by Using the Domain Name System (DNS) ......................................... 7-17
    How a Client Locates a Directory Server by Using DNS ............................................................................... 7-17
    Registering a Directory Server with the Domain Name System ................................................................... 7-18

8 Directory Entries Administration

Managing Entries by Using Oracle Directory Manager .......................................................................................... 8-1
  Searching for Entries by Using Oracle Directory Manager ................................................................................ 8-1
  Viewing Attributes for a Specific Entry by Using Oracle Directory Manager ....................................................... 8-3
  Adding Entries by Using Oracle Directory Manager .......................................................................................... 8-3
    Adding a New Entry by Using Oracle Directory Manager ................................................................................ 8-3
    Adding an Entry by Copying an Existing Entry in Oracle Directory Manager ................................................. 8-4
    Example: Adding a User Entry by Using Oracle Directory Manager .............................................................. 8-4
  Modifying Entries by Using Oracle Directory Manager .................................................................................... 8-5
    Example: Modifying a User Entry by Using Oracle Directory Manager .......................................................... 8-5
  Managing Entries with Attribute Options by Using Oracle Directory Manager ................................................. 8-6
    Adding an Attribute Option to an Existing Entry by Using Oracle Directory Manager .................................. 8-6
    Modifying an Attribute Option by Using Oracle Directory Manager ............................................................ 8-6
    Deleting an Attribute Option by Using Oracle Directory Manager .................................................................. 8-7

Managing Entries by Using Command-Line Tools .................................................................................................. 8-7
  Command-Line Tools for Managing Entries ....................................................................................................... 8-7
    Example: Adding a User Entry by Using ldapadd ............................................................................................ 8-8
    Example: Modifying a User Entry by Using ldapmodify .................................................................................. 8-8
  Managing Entries with Attribute Options by Using Command-Line Tools ........................................................ 8-9
    Example: Adding an Attribute Option by Using ldapmodify ........................................................................... 8-9
    Example: Deleting an Attribute Option by Using ldapmodify ....................................................................... 8-9
    Example: Searching for Entries with Attribute Options by Using ldapsearch .............................................. 8-9

Managing Knowledge References and Referrals ..................................................................................................... 8-10
  Configuring Smart Referrals .................................................................................................................................. 8-10
  Configuring Default Referrals ............................................................................................................................ 8-11
  Client-Side Referral Caching .................................................................................................................................. 8-11
    How Client-Side Referral Caching Works ......................................................................................................... 8-12

9 Using Bulk Tools

bulkload ........................................................................................................................................................................ 9-2
10 Attribute Uniqueness in the Directory

About Attribute Uniqueness ................................................................. 10-1

Rules for Creating Attribute Uniqueness .......................................... 10-2
  Specifying Multiple Attribute Names in an Attribute Uniqueness Constraint ....................................................... 10-3
  Specifying Multiple Subtrees in an Attribute Uniqueness Constraint ................................................................. 10-3
  Specifying Multiple Scopes in an Attribute Uniqueness Constraint ................................................................. 10-4
  Specifying Multiple Object Classes in an Attribute Uniqueness Constraint .......................................................... 10-4
11 Directory Schema Administration

About the Directory Schema ................................................................. 11-1

Object Classes in the Directory .......................................................... 11-2

  About Object Class Management .................................................... 11-2
  Inheritance ..................................................................................... 11-2
  Mandatory and Optional Attributes in Object Classes ..................... 11-2
  Addition of Entries in Top-Down Sequence ..................................... 11-3
  Object Class Explosion .................................................................. 11-3

  Guidelines for Adding, Modifying, and Deleting Object Classes ....... 11-3
  Guidelines for Adding Object Classes ............................................ 11-3
  Guidelines for Modifying Object Classes ....................................... 11-4
  Guidelines for Deleting Object Classes ........................................ 11-5

  Managing Object Classes by Using Oracle Directory Manager ....... 11-5
  Searching for Object Classes by Using Oracle Directory Manager ...... 11-5
  Viewing Properties of Object Classes by Using Oracle Directory Manager .. 11-6
  Adding Object Classes by Using Oracle Directory Manager .......... 11-6
  Modifying Object Classes by Using Oracle Directory Manager ...... 11-6
  Deleting Object Classes by Using Oracle Directory Manager ....... 11-7

  Managing Object Classes by Using Command-Line Tools .............. 11-7
  Example: Adding a New Object Class ........................................... 11-7
  Example: Adding a New Attribute to an Auxiliary or User-Defined Object Class .. 11-8

Attributes in the Directory ................................................................. 11-8

  About Attribute Management ....................................................... 11-9
  Rules for Adding Attributes .......................................................... 11-9
  Rules for Modifying Attributes ..................................................... 11-9
  Rules for Deleting Attributes ....................................................... 11-10

  Managing Attributes by Using Oracle Directory Manager .............. 11-10
  Viewing All Directory Attributes by Using Oracle Directory Manager .. 11-10
  Searching for Attributes by Using Oracle Directory Manager .......... 11-10
  Adding an Attribute by Using Oracle Directory Manager ............... 11-11
12 Referential Integrity

Configuring and Enabling Referential Integrity ................................................................. 12-1
Disabling Referential Integrity .......................................................................................... 12-3

13 Dynamic and Static Groups in Oracle Internet Directory

About Groups ....................................................................................................................... 13-1

Static Groups ..................................................................................................................... 13-1

Schema Elements for Creating Static Groups ................................................................. 13-1

Dynamic Groups .............................................................................................................. 13-2
### 14 Monitoring Oracle Internet Directory Servers

- **Capabilities of Oracle Internet Directory Server Manageability** ........................................ 14-14
- **Oracle Internet Directory Server Manageability Architecture and Components** ............... 14-14
- **Location of Configuration Information for Oracle Internet Directory Server Manageability** . 14-16
- **Account Used for Accessing Server Manageability Information** ....................................... 14-16
- **Configuring Oracle Internet Directory Server Manageability** .......................................... 14-16
  - Configuring Security Events Tracking................................................................................. 14-17
  - Configuring Critical Events............................................................................................... 14-19
- **Purging of Audit and Statistics Entries** ............................................................................. 14-19
- **Viewing Oracle Internet Directory Server Manageability Information** ............................ 14-19
  - Viewing Information with the oiddiag Tool......................................................................... 14-20
  - Viewing Information with the Oracle Identity ManagementGrid Control Plug-in.............. 14-20
  - Viewing Information with the Oracle Enterprise Manager 10g Application Server Control Console........................................................................................................ 14-20

### 15 Backup and Restoration of a Directory

- **Backing Up and Restoring a Small Directory or Specific Naming Context** ....................... 15-1
- **Backing Up and Restoring a Large Directory** .................................................................... 15-1

### Part III Directory Security

### 16 Directory Security Concepts

- **Data Integrity and Oracle Internet Directory** .................................................................... 16-1
- **Data Privacy and Oracle Internet Directory** ..................................................................... 16-1
  - Privacy During Data Transmission.................................................................................... 16-2
  - Privacy of Retrieved Sensitive Attributes....................................................................... 16-2
- **Authorization in Oracle Internet Directory** ..................................................................... 16-3
- **Authentication in Oracle Internet Directory** .................................................................... 16-4
  - Direct Authentication........................................................................................................ 16-4
  - Indirect Authentication..................................................................................................... 16-5
  - External Authentication..................................................................................................... 16-7
- **Protection of User Passwords for Directory Authentication** ........................................... 16-7
- **Password Policies in Oracle Internet Directory** ............................................................... 16-7
- **Authentication by Using Simple Authentication and Security Layer (SASL)** ................... 16-8

### 17 Secure Sockets Layer (SSL) and the Directory

- **Supported Cipher Suites** .................................................................................................. 17-1
- **SSL Client Scenarios** ....................................................................................................... 17-2
- **Limitations of the Use of SSL in 10g (10.1.4.0.1)** ......................................................... 17-2
- **Configuring and Testing Oracle Internet Directory With SSL** ........................................ 17-2
18 Directory Access Control

Overview of Access Control Policy Administration ................................................................. 18-1
   Access Control Management Constructs............................................................................. 18-2
   Access Control Policy Points (ACPs) .................................................................................. 18-2
   The orclACI Attribute for Prescriptive Access Control....................................................... 18-2
   The orclEntryLevelACI Attribute for Entry-Level Access Control.................................... 18-2
   Security Groups..................................................................................................................... 18-3
Access Control Information Components ............................................................................... 18-6
   Object: To What Are You Granting Access? ....................................................................... 18-6
   Subject: To Whom Are You Granting Access? ..................................................................... 18-7
Access Level Requirements for LDAP Operations ................................................................. 18-10

How ACL Evaluation Works ................................................................................................... 18-10
   Precedence Rules Used in ACL Evaluation......................................................................... 18-11
       Precedence at the Entry Level ......................................................................................... 18-11
       Precedence at the Attribute Level ................................................................................... 18-12
   Use of More Than One ACI for the Same Object ............................................................... 18-12
   Exclusionary Access to Directory Objects ......................................................................... 18-13
   ACL Evaluation For Groups ............................................................................................... 18-13

Managing Access Control by Using Oracle Directory Manager ................................................ 18-14
   Configuring Oracle Directory Manager for Access Control Management....................... 18-14
       Configuring the Display of ACPs in Oracle Directory Manager .................................... 18-14
       Configuring Searches for ACPs When Using Oracle Directory Manager ....................... 18-15
   Viewing an ACP by Using Oracle Directory Manager ......................................................... 18-15
   Adding an ACP by Using Oracle Directory Manager ............................................................ 18-16
       Task 1: Specify the Entry That Will Be the ACP ............................................................... 18-16
       Task 2: Configure Structural Access Items .................................................................... 18-16
       Task 3: Configure Content Access Items ...................................................................... 18-17
   Adding an ACP by Using the ACP Creation Wizard of Oracle Directory Manager ............... 18-18
       Task 1: Specify the Entry That Will Be the ACP ............................................................... 18-19
       Task 2: Configure Structural Access Items by Using the ACP Creation Wizard ............. 18-19
       Task 3: Configure Content Access Items by Using the ACP Creation Wizard ............... 18-20
   Modifying an ACP by Using Oracle Directory Manager ...................................................... 18-21
       Task 1: Specify the Entry That You Want to Modify ......................................................... 18-21
       Task 2: Modify Structural Access Items ........................................................................... 18-22
       Task 3: Modify Content Access Items .............................................................................. 18-23
19  Password Policies in Oracle Internet Directory

About Password Policies........................................................................................................ 19-1
What a Password Policy Is.................................................................................................. 19-1
Fine-Grained Password Policies ........................................................................................ 19-2
Default Password Policy ..................................................................................................... 19-4
Password Policy Attributes ................................................................................................. 19-5
Directory Server Verification of Password Policy Information........................................... 19-7

Managing Password Policies, Accounts, and Passwords ................................................. 19-8
Managing Password Policies by Using Oracle Directory Manager.................................... 19-8
Viewing Password Policies by Using Oracle Directory Manager .................................... 19-9
Modifying Password Policies by Using Oracle Directory Manager .................................. 19-9
Creating Password Policies by Using Oracle Directory Manager .................................... 19-9
Example: Setting Password Policies by Using Command-Line Tools ................................ 19-10
Examples: Managing Password Policies by Using Command-Line Tools .......................... 19-10
Example: Enabling and Disabling Accounts by Using Command-Line Tools .................... 19-11
Example: Unlocking Accounts by Using Command-Line Tools ....................................... 19-11
Example: Forcing a Password Change by Using Command-Line Tools ............................ 19-12
Managing Accounts and Passwords by Using the Self-Service Console ....................... 19-12
Enabling and Disabling Accounts by Using the Oracle Internet Directory Self-Service Console ........................................................................................................................................ 19-12
Unlocking Accounts by Using the Oracle Internet Directory Self-Service Console .......... 19-12
Resetting Your Own Password by Using the Oracle Internet Directory Self-Service Console .......................................................................................................................... 19-12
Password Policy Error Messages......................................................................................... 19-13

20  Directory Storage of Password Verifiers

About Centralized Storage of User Authentication Credentials ........................................ 20-1
Storing and Managing Password Verifiers for Authenticating to Oracle Internet Directory .................................................. 20-1
Password Verifiers and Authentication to the Directory.................................................... 20-2
21 Delegation of Privileges for an Oracle Technology Deployment

21-1 Delegation in the Oracle Identity Management Model

21-2 How Delegation Works

21-3 Delegation in the Oracle Application Server Environment

21-3 About the Default Configuration

21-4 Overview: Privileges for Administering the Oracle Technology Stack

21-4 Delegation of Privileges for User and Group Management

21-5 How Privileges Are Granted for Managing User and Group Data

21-5 Default Privileges for Managing User Data

21-5 Creating Users for a Realm

21-6 Modifying Attributes of a User

21-6 Deleting a User

21-6 Delegating User Administration

21-7 Default Privileges for Managing Group Data

21-7 Creating Groups

21-7 Modifying the Attributes of Groups

21-7 Deleting Groups

21-8 Delegating Group Administration

21-8 Delegation of Privileges for Deployment of Oracle Components

21-9 How Deployment Privileges Are Granted

21-9 Oracle Application Server Administrators

21-10 User Management Application Administrators

21-10 Trusted Application Administrators

21-10 Delegation of Privileges for Component Runtime

21-11 Default Privileges for Reading and Modifying User Passwords

21-12 Default Privileges for Comparing User Passwords

21-12 Default Privileges for Comparing Password Verifiers
<table>
<thead>
<tr>
<th>Default Privileges for Proxying on Behalf of End Users</th>
<th>21-13</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Privileges for Managing the Oracle Context</td>
<td>21-13</td>
</tr>
<tr>
<td>Default Privileges for Reading Common User Attributes</td>
<td>21-13</td>
</tr>
<tr>
<td>Default Privileges for Reading Common Group Attributes</td>
<td>21-14</td>
</tr>
<tr>
<td>Default Privileges for Reading the Service Registry</td>
<td>21-14</td>
</tr>
<tr>
<td>Default Privileges for Administering the Service Registry</td>
<td>21-14</td>
</tr>
</tbody>
</table>

**Part IV Directory Deployment**

**22 Directory Deployment Considerations**

The Expanding Role of Directories ................................................................. 22-1
Logical Organization Of Directory Information ............................................. 22-2
Physical Distribution: Partitions, Replicas, and High Availability ............. 22-2
   An Ideal Deployment ...................................................................................... 22-2
   Partitioning Considerations ....................................................................... 22-3
   Replication Considerations ......................................................................... 22-3
   High Availability Considerations ............................................................. 22-4
Oracle Directory Integration Platform ......................................................... 22-5
Capacity Planning, Sizing, and Tuning ......................................................... 22-6
   Capacity Planning ........................................................................................ 22-6
   Sizing Considerations .................................................................................. 22-6
   Tuning Considerations ................................................................................ 22-8

**23 Deployment of Oracle Identity Management Realms**

Planning the Directory Information Tree for Identity Management ................ 23-1
Planning the Overall Directory Structure ..................................................... 23-3
Planning the Names and Containment of Users and Groups .......................... 23-3
   Considerations for Users ........................................................................... 23-4
   Considerations for Groups ......................................................................... 23-4
Planning the Identity Management Realm .................................................... 23-5
Migrating a DIT from a Third-Party Directory .............................................. 23-7
Identity Management Realms in an Enterprise Deployment .......................... 23-7
   Single Identity Management Realm in the Enterprise ............................... 23-7
   Multiple Identity Management Realms in the Enterprise .......................... 23-8
Identity Management Realms in a Hosted Deployment .................................. 23-8
Identity Management Realm Implementation in Oracle Internet Directory .... 23-9
Default Directory Information Tree and the Identity Management Realm ....... 23-10
Administration of Identity Management Realms ......................................... 23-12
   Customizing the Default Identity Management Realm ............................. 23-12
   Changing the Location of Users and Groups In The Default Identity Management Realm .................................................. 23-13
   Creating Additional Identity Management Realms for Hosted Deployments .. 23-18

**24 Capacity Planning for the Directory**

About Capacity Planning ................................................................................... 24-1
Getting to Know Directory Usage Patterns: A Case Study .............................. 24-2
I/O Subsystem Requirements................................................................. 24-4
  About the I/O Subsystem........................................................................ 24-5
  Rough Estimates of Disk Space Requirements ......................................... 24-5
  Detailed Calculations of Disk Space Requirements .................................... 24-6
Memory Requirements.............................................................................. 24-9
Network Requirements............................................................................ 24-10
CPU Requirements................................................................................. 24-11
  CPU Configuration..................................................................................... 24-11
  Rough Estimates of CPU Requirements .................................................. 24-11
  Detailed Calculations of CPU Requirements ............................................. 24-12
Summary of Capacity Plan for Acme Corporation........................................ 24-13

25 Tuning Considerations for the Directory

About Tuning .......................................................................................... 25-1
Tools for Performance Tuning ................................................................. 25-2
CPU Usage Tuning ............................................................................... 25-2
  Tuning CPU for Oracle Internet Directory Processes................................. 25-3
  Tuning CPU for Oracle Foreground Processes............................................ 25-4
  Taking Advantage of Processor Affinity on SMP Systems ......................... 25-4
  Other Alternatives for a CPU Constrained System ..................................... 25-5
Memory Tuning ...................................................................................... 25-5
  Tuning the System Global Area (SGA) for the Oracle Database.................. 25-5
  Other Alternatives for a Memory-Constrained System............................... 25-5
  Tuning Security Event Tracking ............................................................... 25-6
  Tuning Memory Allocated for Event Tracking ............................................ 25-6
  Tuning Memory Used for Each Operation .................................................. 25-6
Disk Tuning ......................................................................................... 25-6
Database Tuning .................................................................................... 25-7
  Required Parameters .................................................................................. 25-7
  Parameters Dependent on Oracle Internet Directory Server Configuration.... 25-8
    Using Shared Server Process ................................................................. 25-8
  SGA Parameters Dependent on Hardware Resources ............................... 25-8
Entry Caching ....................................................................................... 25-8
Caching of Connection DNs ................................................................... 25-9
Optimizing Searches ............................................................................ 25-9
  Optimizing Subtree Searches ................................................................. 25-9
  Optimizing Searches for Large Group Entries ........................................... 25-9
    Entry Cache Enabled Configuration ....................................................... 25-10
    Entry Cache Disabled Configuration ...................................................... 25-10
  Optimizing Searches for Skewed Attributes ............................................. 25-10
    Optimizing Searches for Skewed Attributes by Using Oracle Directory Manager ... 25-11
    Optimizing Searches for Skewed Attributes by Using ldapmodify ............ 25-11
Setting the Time Limit Mode ................................................................. 25-11
  Setting the Time Limit Mode by Using Oracle Directory Manager ................ 25-11
  Setting the Time Limit Mode by Using ldapmodify .................................... 25-12
Setting the Timeout for Client/Server Connections ................................... 25-12
Setting the Timeout for Write Operations .............................................. 25-12
26  Garbage Collection in Oracle Internet Directory

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>About the Oracle Internet Directory Garbage Collection Framework</td>
<td>26-1</td>
</tr>
<tr>
<td>Components of the Oracle Internet Directory Garbage Collection Framework</td>
<td>26-1</td>
</tr>
<tr>
<td>Garbage Collection Plug-in</td>
<td>26-1</td>
</tr>
<tr>
<td>Background Database Processes</td>
<td>26-2</td>
</tr>
<tr>
<td>How Oracle Internet Directory Garbage Collection Works</td>
<td>26-4</td>
</tr>
<tr>
<td>Garbage Collector Entries and the Oracle Internet Directory Statistics Collector Entry</td>
<td>26-5</td>
</tr>
<tr>
<td>Change Log Purging in Multimaster Replication</td>
<td>26-5</td>
</tr>
</tbody>
</table>

**Modifying Oracle Internet Directory Garbage Collectors**

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modifying a Garbage Collector by Using Oracle Directory Manager</td>
<td>26-6</td>
</tr>
<tr>
<td>Modifying a Garbage Collector by Using Command-Line Tools</td>
<td>26-7</td>
</tr>
<tr>
<td>Example 1: Modifying a Garbage Collector</td>
<td>26-7</td>
</tr>
<tr>
<td>Example 2: Disabling a Garbage Collector Change Log</td>
<td>26-7</td>
</tr>
<tr>
<td>Modifying the Oracle Internet Directory Statistics Collector</td>
<td>26-7</td>
</tr>
</tbody>
</table>

**Enabling, Disabling, and Monitoring Logging for Oracle Internet Directory Garbage Collectors**

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabling Logging for Oracle Internet Directory Garbage Collectors</td>
<td>26-7</td>
</tr>
<tr>
<td>Disabling Logging for Oracle Internet Directory Garbage Collectors</td>
<td>26-8</td>
</tr>
<tr>
<td>Monitoring Garbage Collection Logging</td>
<td>26-8</td>
</tr>
</tbody>
</table>

27  Migration of Data from Other Data Repositories

**The Default Directory Structure of Oracle Internet Directory**

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Migrating Data from LDAP-Compliant Directories</td>
<td>27-1</td>
</tr>
<tr>
<td>Tools</td>
<td>27-2</td>
</tr>
<tr>
<td>bulkload</td>
<td>27-2</td>
</tr>
<tr>
<td>dipassistant</td>
<td>27-2</td>
</tr>
<tr>
<td>Oracle Directory Integration Platform Server</td>
<td>27-3</td>
</tr>
<tr>
<td>Common Usage Scenarios</td>
<td>27-3</td>
</tr>
<tr>
<td>Scenario 1: Using an LDIF File and bulkload</td>
<td>27-3</td>
</tr>
<tr>
<td>Scenario 2: Using dipassistant Directly</td>
<td>27-4</td>
</tr>
<tr>
<td>Scenario 3: Using an LDIF File and dipassistant</td>
<td>27-4</td>
</tr>
<tr>
<td>Scenario 4: Using dipassistant, bulkload, and LDIF Files</td>
<td>27-5</td>
</tr>
<tr>
<td>Scenario 5: Using the Oracle Directory Integration Platform Server</td>
<td>27-6</td>
</tr>
<tr>
<td>Tasks For Migrating Data from LDAP-Compliant Directories</td>
<td>27-6</td>
</tr>
<tr>
<td>Task 1: Export Data from the Non-Oracle Internet Directory Server into LDIF File Format</td>
<td>27-6</td>
</tr>
<tr>
<td>Task 2: Analyze the LDIF User Data for Any Required Schema Additions</td>
<td>27-6</td>
</tr>
<tr>
<td>Task 3: Extend the Schema in Oracle Internet Directory</td>
<td>27-6</td>
</tr>
<tr>
<td>Task 4: Remove Any Proprietary Directory Data from the LDIF File</td>
<td>27-7</td>
</tr>
<tr>
<td>Task 5: Remove Operational Attributes from the LDIF File</td>
<td>27-7</td>
</tr>
<tr>
<td>Task 6: Remove Incompatible userPassword Attribute Values from the LDIF File</td>
<td>27-7</td>
</tr>
<tr>
<td>Task 7: Run the bulkload check=&quot;TRUE&quot; Mode and Determine Any Remaining Schema Violations or Duplication Errors</td>
<td>27-8</td>
</tr>
</tbody>
</table>

**Migrating User Data from Application-Specific Repositories**

<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Intermediate Template File</td>
<td>27-8</td>
</tr>
</tbody>
</table>
Reconciling Data in Application Repository with Data Already in Oracle Internet Directory .......................................................... 27-9
Tasks For Migrating Data from Application-Specific Repositories .......................................................... 27-9
  Task 1: Create an Intermediate Template File ................................................................................. 27-9
  Task 2: Run the OID Migration Tool ............................................................................................ 27-11

28 Server Chaining

Supported External Servers ............................................................................................................ 28-1
Integrated Oracle Products .......................................................................................................... 28-2
Supported Operations .................................................................................................................. 28-2
Server Chaining with Replication ............................................................................................... 28-3
Configuring Server Chaining ....................................................................................................... 28-3
  Configuring Server Chaining from the Command Line ............................................................. 28-3
  Configuring Server Chaining by Using Oracle Directory Manager ........................................ 28-4
Requirements for User and Group Containers ........................................................................... 28-4
Attribute Mapping ....................................................................................................................... 28-5
Server Chaining Configuration Entries ....................................................................................... 28-5
  Configuration Entry Attributes ............................................................................................... 28-6
  Active Directory Example ......................................................................................................... 28-7
  Sun Java System Directory Server (iPlanet) Example .............................................................. 28-8
Debugging Server Chaining ......................................................................................................... 28-9

Part V Directory Replication

29 Oracle Internet Directory Replication Concepts

Replication Concepts .................................................................................................................. 29-1
  Content to be Replicated: Full or Partial .................................................................................. 29-2
  Direction: One-Way or Two-Way ............................................................................................ 29-2
  Transport Mechanism: Advanced Database Replication or LDAP ........................................ 29-3
Directory Replication Group (DRG) Types .................................................................................. 29-3
Directory Replication Groups ....................................................................................................... 29-4
  Data Transfer Between Nodes in a Directory Replication Group ........................................... 29-4
  Single-Master Replication Groups .......................................................................................... 29-5
  Multimaster Replication Groups ............................................................................................. 29-5
  Fan-Out Replication Groups .................................................................................................... 29-6
  Types of Directory Replication Compared ............................................................................ 29-6
  Multimaster Replication with Fan-Out .................................................................................... 29-7
Replication Configuration Objects in the Directory ..................................................................... 29-8
  The Replication Configuration Container .............................................................................. 29-9
  The Replica Subentry ............................................................................................................... 29-9
  The Replication Agreement Entry .......................................................................................... 29-10
    Replication Agreement Entry Attributes ............................................................................ 29-10
    Advanced Replication Agreements ...................................................................................... 29-11
    LDAP Replication Agreements ............................................................................................. 29-12
    Two-Way LDAP Replication Agreements .......................................................................... 29-12
  The Replication Naming Context Container Entry ................................................................... 29-13
The Replication Naming Context Object Entry ................................................................. 29-13
Directory Replication Server Configuration Parameters .................................................. 29-14
Examples of Replication Configuration Objects in the Directory ..................................... 29-15
Replication Security .......................................................................................................... 29-18
Authentication and the Directory Replication Server ....................................................... 29-18
Secure Sockets Layer (SSL) and Oracle Internet Directory Replication ......................... 29-19
Change Logs in Directory Replication .............................................................................. 29-19
Oracle Database Advanced Replication ............................................................................. 29-20
Features of Oracle Database Advanced Replication ....................................................... 29-20
Architecture for Oracle Advanced Database Replication .............................................. 29-21
LDAP-Based Replication ................................................................................................. 29-23
Conflict Resolution in Oracle Replication .......................................................................... 29-25
Levels at Which Replication Conflicts Occur ..................................................................... 29-25
Typical Causes of Conflicts ............................................................................................... 29-26
Automated Resolution of Conflicts .................................................................................. 29-26
Replication Failover .......................................................................................................... 29-27
Included and Excluded Naming Contexts in Partial Replication ....................................... 29-30
Oracle Database Advanced Replication Filtering ............................................................ 29-30
LDAP Replication Filtering ............................................................................................... 29-31
Rules for LDAP Replication Filtering ............................................................................. 29-31
Examples of LDAP Replication Filtering ....................................................................... 29-31
Rules for Managing Naming Contexts and Attributes .................................................... 29-35
Optimization of Partial Replication Naming Context for Better Performance ................ 29-36

30 Oracle Internet Directory Replication Installation and Configuration

Oracle Internet Directory Versions and Replication .......................................................... 30-1
Preliminary Information for Installing and Configuring a Replication Group ................... 30-2
Oracle Internet Directory Installation ................................................................................. 30-2
If You are Installing Oracle Internet Directory as a Master ............................................ 30-3
If You are Installing Oracle Internet Directory as an Advanced Replication-Based Replica or as a One-Way or Two-Way LDAP-Based Replica ................................................. 30-3
The Replication Environment Management Tool ........................................................... 30-4
Installing and Configuring Multimaster Replication ........................................................ 30-5
Rules for Configuring Directory Replication Based on Oracle Database Advanced Replication ...................................................................................................................... 30-5
Installing and Configuring a Multimaster Replication Group .......................................... 30-6
Task 1: Install Oracle Internet Directory as a Master on the Master Definition Site (MDS) ......................................................................................................................... 30-7
Task 2: Install the Oracle Internet Directory as a Replica, on the Remote Master Sites (RMS) ..................................................................................................................... 30-7
Task 3: Set Up Oracle Database Advanced Replication for a Directory Replication Group ......................................................................................................................... 30-9
Task 4 (Optional): Load Data into the Directory ............................................................... 30-12
Task 5: Ensure that Oracle Directory Server Instances are Started on All the Nodes . 30-13
Task 6: Start the Replication Servers on All Nodes in the DRG ........................................ 30-13
Task 7: Test Directory Replication .................................................................................. 30-14
Adding a Node for Multimaster Replication (Oracle Database Advanced Replication Types Only)........................................................................................................... 30-14
Prepare the Oracle Net Services Environment ........................................................................ 30-15
Task 1: Stop the Directory Replication Server on All Nodes.................................................... 30-15
Task 2: Identify a Sponsor Node and Install Oracle Internet Directory as a Replica on the Remote Site .................................................................................................................. 30-15
Task 3: Switch the Sponsor Node to Read-Only Mode ............................................................. 30-15
Task 4: Back up the Sponsor Node by Using ldifwrite............................................................. 30-16
Task 5: Perform Advanced Replication Add Node Setup ....................................................... 30-16
Task 6: Switch the Sponsor Node to Updatable Mode ............................................................. 30-17
Task 7: Start the Directory Replication Server on All Nodes Except the New Node.............. 30-17
Task 8: Load Data into the New Node by Using bulkload....................................................... 30-18
Task 9: Start the Directory Server on the New Node............................................................... 30-18
Task 10: Start the Directory Replication Server on the New Node...................................... 30-18
Deleting a Node from a Multimaster Replication Group .......................................................... 30-18
Task 1: Stop the Directory Replication Server on All Nodes.................................................... 30-19
Task 2: Stop All Oracle Internet Directory Processes in the Node to be Deleted ................. 30-19
Task 3: Delete the Node from the Master Definition Site ....................................................... 30-19
Task 4: Start the Directory Replication Server on All Nodes .................................................. 30-19
Installing and Configuring One-Way or Two-Way LDAP-Based Replication ....................... 30-19
Rules for Configuring LDAP-Based Replication .................................................................. 30-20
Back Up Your LDAP Data by Using ldifwrite and bulkload ................................................. 30-21
Installing and Configuring a One-Way or Two-Way LDAP Replica with Default Settings .... 30-21
Task 1: Identify and Start the Directory Server on the Supplier Node .................................... 30-21
Task 2: Installing Oracle Internet Directory As An LDAP Replica ....................................... 30-22
Task 3: Ensure the Directory Replication Servers are Started .............................................. 30-22
Installing and Configuring an LDAP-Based Replica with Customized Settings ................... 30-22
Configuring an LDAP-Based Replica by Using Automatic Bootstrapping ......................... 30-23
Configuring an LDAP-Based Replica by Using the ldifwrite Tool .................................... 30-28
Password Policy and Fan-out Replication .............................................................................. 30-32
Deleting an LDAP-Based Replica ............................................................................................ 30-33
Task 1: Stop the Directory Replication Server on the Node to be Deleted ......................... 30-33
Task 2: Delete the Replica from the Replication Group ......................................................... 30-33
Task 3: Stop the Directory Server on the Node to be Deleted .............................................. 30-33
Determining What Is to Be Replicated in LDAP-Based Partial Replication .......................... 30-33
Viewing and Modifying Replica Naming Context Objects by Using Oracle Directory Manager ................................................................................................................................. 30-34
Adding Replica Naming Context Objects by Using Oracle Directory Manager .................... 30-34
Deleting Replica Naming Context Objects by Using Oracle Directory Manager ................... 30-34
Modifying Replica Naming Context Object Parameters by Using ldapmodify ..................... 30-34
Resolving Conflicts Manually in a Replication Group ............................................................ 30-37
Monitoring Replication Change Conflicts ............................................................................. 30-37
Examples of Conflict Resolution Messages ......................................................................... 30-37
About the Human Intervention Queue Manipulation Tool ..................................................... 30-38
About the Oracle Internet Directory Comparison and Reconciliation Tool .......................... 30-38
Example: Installing and Configuring a Multimaster Replication Group with Fan-Out ........... 30-39
Configuring Replication Failover ........................................................................................... 30-43
Limitations and Warnings for Replication Failover ................................................................. 30-43
Determining Which Type of Replication Failover to Use .................................................... 30-44
Performing a Stateless Replication Failover ................................................................. 30-44
  Task 1: Stop all Directory Replication Server on related Nodes .................. 30-44
  Task 2: Break Old Replication Agreement and Setup New Agreement .......... 30-45
  Task 3: Save Last Change Number ..................................................................... 30-45
  Task 4: Compare and Reconcile New Supplier and Consumer ..................... 30-45
  Task 5: Update Last Applied Change Number of New Agreement ............... 30-46
  Task 6: Clean Up Old Agreement on Old Supplier ........................................... 30-46
  Task 7: Start all Directory Replication Server on related Nodes .................. 30-47
Performing a Time-Based Replication Failover ....................................................... 30-47
  Task 1: Configure Change Log Garbage Collection Object on New Supplier .... 30-47
  Task 2: Save Last Change Number from New Supplier ..................................... 30-47
  Task 3: Enable Change Log Regeneration on New Supplier ......................... 30-48
  Task 4: Wait for the Desired Time Period to Elapse ............................................. 30-48
  Task 5: Stop all Directory Replication Servers on Related Nodes ................ 30-48
  Task 6: Break Old Replication Agreement and Set Up New Agreement .......... 30-48
  Task 7: Update Last Applied Change Number of New Agreement ............... 30-48
  Task 8: Clean Up Old Agreement on Old Supplier ........................................... 30-49
  Task 9: Start All Directory Replication Servers on Related Nodes ............... 30-49

31 Oracle Internet Directory Replication Monitoring and Management

Viewing and Modifying Directory Replication Server Configuration Parameters .......... 31-1
  Viewing Configuration Parameters of the Directory Replication Server by Using
  Oracle Directory Manager ...................................................................................... 31-2
  Modifying Configuration Parameters of the Directory Replication Server by Using
  Oracle Directory Manager ...................................................................................... 31-2
  Modifying Directory Replication Server Configuration Parameters by Using
  Command-Line Tools ............................................................................................. 31-2

Viewing and Modifying Parameters for Particular Replica Nodes ......................... 31-4
  Viewing and Modifying Parameters for a Particular Replica Node by Using Oracle
  Directory Manager ............................................................................................... 31-4
  Modifying a Particular Replica Node by Using Command-Line Tools ............... 31-4

Modifying Parameters for Replication Agreements ............................................. 31-6
  Modifying Parameters for Replication Agreements Based on Oracle Database Advanced
  Replication ............................................................................................................ 31-6
    Viewing and Modifying Replication Agreements Based on Oracle Database
    Advanced Replication by Using Oracle Directory Manager ............................. 31-6
    Managing Replication Agreements Based on Advanced Replication by Using
    Idapmodify ............................................................................................................ 31-7
  Modifying Parameters for Replication Agreements Based on LDAP .................... 31-8
    Viewing and Modifying LDAP-Based Replication Agreement Parameters by Using
    Oracle Directory Manager .................................................................................. 31-8
    Modifying LDAP-Based Replication Agreement Parameters by Using Idapmodify .... 31-9

Changing the Replication Administrator's Password on All Nodes Using Oracle Database
Advanced Replication ............................................................................................... 31-10
Managing the Change Log ...................................................................................... 31-10
Modifying the Speed of Directory Replication ...................................................... 31-11
Modifying the Speed of Directory Replication When Using Oracle Database Advanced Replication ................................................................. 31-11

Modifying the Speed of Directory Replication When Using LDAP-Based Replication ................................................................. 31-11

Managing and Monitoring Topology .............................................................................................................................. 31-12

The Compare and Reconcile Tool ........................................................................................................................................... 31-12

Conflict Scenarios .................................................................................................................................................................................. 31-13
Operations Supported by oidcmprec .................................................................................................................................................. 31-13
Output from oidcmprec .............................................................................................................................................................................. 31-14
How oidcmprec Works ............................................................................................................................................................................... 31-14
Setting the Source and Destination Directories .............................................................................................................................. 31-15
Selecting the DIT for the Operation .................................................................................................................................................. 31-15
Selecting the Attributes for the Operation ........................................................................................................................................ 31-16
Controlling Change Log Generation .................................................................................................................................................. 31-16
Using a Parameter File ............................................................................................................................................................................. 31-17
Including Directory Schema ...................................................................................................................................................................... 31-17
Overriding Predefined Conflict Resolution Rules .............................................................................................................................. 31-18
Using the User-Defined Compare and Reconcile Operation ................................................................................................................... 31-18
Known Limitations of the oidcmprec Tool ........................................................................................................................................ 31-18

Part VI Directory Plug-ins

32 Oracle Internet Directory Server Plug-in Framework

About Directory Server Plug-ins .............................................................................................................................................................. 32-1
LDAP Operations and Timings Supported by the Directory .............................................................................................................. 32-2
Pre-Operation Server Plug-ins ............................................................................................................................................................... 32-3
Post-Operation Server Plug-ins ........................................................................................................................................................... 32-3
When-Operation Server Plug-ins ......................................................................................................................................................... 32-3
When_Replace-Operation Server Plug-ins ........................................................................................................................................ 32-3

Creating Plug-ins .................................................................................................................................................................................... 32-4

Registering and Managing Plug-ins ...................................................................................................................................................... 32-4
Registering and Managing Plug-ins by Using Oracle Directory Manager ......................................................................................................................... 32-5
Adding a Plug-in Configuration Entry by Using Oracle Directory Manager .......................................................................................... 32-5
Editing a Plug-in by Using Oracle Directory Manager ...................................................................................................................... 32-5
Deleting a Plug-in by Using Oracle Directory Manager ..................................................................................................................... 32-5
Registering and Managing Plug-ins by Using Command-Line Tools ........................................................................................................... 32-6
Example: Adding a Plug-in Configuration Entry by Using Command-Line Tools .................................................................................. 32-6
Example: Modifying a Plug-in Configuration Entry by Using Command-Line Tools ............................................................................... 32-7
Example: Deleting a Plug-in Configuration Entry by Using Command-Line Tools .................................................................................. 32-7

33 Oracle Internet Directory Plug-In for Password Policies

How the Password Policy Plug-in Works ............................................................................................................................................. 33-1
Example: Installing, Configuring, and Enabling a Customized Password Policy Plug-in ........................................................................... 33-2
Loading and Registering the PL/SQL Program ................................................................................................................................... 33-2
Coding the Password Policy Plug-in ..................................................................................................................................................... 33-3
Debugging the Password Policy Plug-in ........................................................................................................................................... 33-3
Contents of Sample PL/SQL Package pluginpkg.sql ....................................................................................................................... 33-4
34 Setting Up the Customized External Authentication Plug-in

Native Authentication Contrasted with External Authentication ................................................ 34-1
Example: Installing, Configuring, and Enabling the External Authentication Plug-in .......... 34-1
Sample PL/SQL Package oidexaup.sql ....................................................................................... 34-2
Debugging the External Authentication Plug-in ......................................................................... 34-3
Contents of PL/SQL Package oidexaup.sql ............................................................................... 34-4

Part VII Appendixes

A Windows and Fields in Oracle Directory Manager

Connection Management Fields in Oracle Directory Manager .................................................. A-2
Access Control Management Fields in Oracle Directory Manager ......................................... A-3
Attribute Uniqueness Fields in Oracle Directory Manager ...................................................... A-5
Garbage Collection Management Fields in Oracle Directory Manager .................................. A-6
Oracle Internet Directory Statistics Collection Management Fields in Oracle Directory Manager ................................................................. A-6
Password Policy Fields in Oracle Directory Manager .............................................................. A-6
Password Verifier Fields in Oracle Directory Manager ............................................................ A-8
Plug-in Management Fields in Oracle Directory Manager ......................................................... A-9
Replication Fields in Oracle Directory Manager ....................................................................... A-12
Schema Management Fields in Oracle Directory Manager ..................................................... A-17
Object Classes Fields in Oracle Directory Manager ............................................................... A-17
Attributes Fields in Oracle Directory Manager ........................................................................ A-19
Matching Rules Fields in Oracle Directory Manager .............................................................. A-22
Content Rules Management Fields in Oracle Directory Manager ........................................... A-22
Server Management Fields in Oracle Directory Manager ...................................................... A-24
Configuration Sets Fields in Oracle Directory Manager ........................................................ A-24
System Operational Attributes Fields in Oracle Directory Manager ...................................... A-25
Super, Guest, and Proxy User Fields in Oracle Directory Manager ......................................... A-29
Query Optimization Fields in Oracle Directory Manager ....................................................... A-29
Entry Search Fields and Buttons in Oracle Directory Manager ............................................. A-31
SSL Management Fields in Oracle Directory Manager .......................................................... A-32
Synchronization Fields in Oracle Directory Manager ............................................................... A-33
Server Chaining Management ................................................................................................. A-35

B The LDAP Filter Definition

C The Access Control Directive Format

Schema for orclACI .................................................................................................................. C-1
Schema for orclEntryLevelACI ............................................................................................... C-2

D Globalization Support in the Directory

About Character Sets and the Directory .................................................................................. D-1
About Unicode......................................................................................................................... D-2
About Oracle and UTF-8.......................................................................................................... D-2
Migration from UTF8 to AL32UTF8 when Upgrading Oracle Internet Directory ................. D-2
The NLS_LANG Environment Variable ........................................... D-3
Using Non-AL32UTF8 Databases .............................................. D-4
Using Globalization Support with LDIF Files .......................... D-4
   An LDIF file Containing Only ASCII Strings ...................... D-4
   An LDIF file Containing UTF-8 Encoded Strings ................. D-4
   CASE 1: Native Strings (Non-UTF-8) ................................. D-5
   CASE 2: UTF-8 Strings ..................................................... D-5
   CASE 3: BASE64 Encoded UTF-8 Strings ......................... D-5
   CASE 4: BASE64 Encoded Native Strings ......................... D-5
   Specifying the -E Argument When Using Each Tool .............. D-6
   Examples: Using the -E Argument with Command-Line LDAP Tools D-6
Setting NLS_LANG in the Client Environment ......................... D-7
Using Globalization Support with Bulk Tools ...................... D-8
   Using Globalization Support with bulkload ...................... D-8
   Using Globalization Support with ldifwrite ..................... D-8
   Using Globalization Support with bulkdelete ................... D-9
   Using Globalization Support with bulkmodify ................... D-9

E Setting up Access Controls for Creation and Search Bases for Users and Groups

   Setting up Access Controls for the User Search Base and the User Creation Base E-1
   Setting up Access Controls for the Group Search Base and the Group Creation Base E-3

F The Multimaster Replication Process

   How the Multimaster Replication Process Adds a New Entry to a Consumer F-1
   How the Multimaster Replication Process Deletes an Entry .................. F-2
   How the Multimaster Replication Process Modifies an Entry ............. F-2
   How the Multimaster Replication Process Modifies a Relative Distinguished Name F-3
   How the Multimaster Replication Process Modifies a Distinguished Name .......... F-4

G Searching the Directory for User Certificates

   Certificate Mapping ......................................................... G-1
   Search Types ......................................................................... G-2

H LDAP Replica States

I Addition of a Directory Node by Using the Database Copy Procedure

   Definitions ........................................................................... I-1
   Prerequisites .......................................................................... I-2
   Sponsor Directory Site Environment .................................. I-2
   New Directory Site Environment ........................................ I-2
   Preliminary Tasks To Be Performed on the New Node ................ I-3
   Addition of an Oracle Advanced Database Replication-Based Directory Node I-3
      Tasks To Be Performed on the Sponsor Advanced Replication Node I-3
Tasks To Be Performed on the New Advanced Replication Node ............................................. I-9
Verification of an Advanced Replication-Based Replica Node ................................................ I-14
Addition of an LDAP Replication-Based Directory Node ........................................................ I-15
Tasks To Be Performed on the Sponsor LDAP Replication Node ............................................. I-15
Tasks To Be Performed on the New LDAP Replication Node .................................................. I-20
Verification of an LDAP-Based Replica Node .......................................................................... I-25

J UNIX Authentication and User Provisioning with Oracle Internet Directory

Schema Customization ................................................................................................................ J-1
UID Attribute Issues ................................................................................................................ J-1

K RFCs Supported by Oracle Internet Directory

L Troubleshooting Oracle Internet Directory

Problems and Solutions ............................................................................................................. L-1
Installation Errors .................................................................................................................... L-1
TCP/IP Problems ...................................................................................................................... L-1
  Do Not Use TCP-Based Monitoring of Oracle Internet Directory Server Availability on Microsoft Windows 2003 Server ................................................................. L-2
  Do Not Install DiamondCS Port Explorer ............................................................................. L-2
Directory Server Error Messages and Causes .......................................................................... L-2
  Oracle Database Server Error Due to Interrupted Client Connection ................................ L-2
  Oracle Database Server Error Due to Schema Modifications ............................................. L-2
  Constraint Violation Error Due to Editing a User or Group or Creating a Realm ................ L-2
  Standard Error Messages Returned from Oracle Directory Server ................................ L-3
  Additional Directory Server Error Messages ...................................................................... L-5
Troubleshooting Password Policies ........................................................................................... L-7
  Password Policy Error Messages ....................................................................................... L-7
Troubleshooting Directory Performance .................................................................................... L-8
  Poor LDAP Search Performance ....................................................................................... L-8
  Poor LDAP Add or Modify Performance .......................................................................... L-9
Troubleshooting Starting, Stopping, and Restarting of the Directory Server ....................... L-9
  About the Tools for Starting, Stopping, and Restarting the Directory Server Instance ...... L-9
  Problems Starting, Stopping, and Restarting the Directory Server ................................ L-11
Troubleshooting Oracle Internet Directory Replication .............................................................. L-14
  Replication Server Does Not Start .................................................................................... L-15
  Repository Creation Assistant Error .................................................................................. L-16
  Errors in Replication Bootstrap ....................................................................................... L-16
  Changes Are Not Replicated ............................................................................................. L-18
  Replication Stops Working ............................................................................................... L-19
Troubleshooting SSL Setup ..................................................................................................... L-20
Troubleshooting Change Log Garbage Collection .................................................................. L-20
  Change Logs Are Not Purged ............................................................................................. L-20
Troubleshooting Dynamic Password Verifiers ...................................................................... L-21
Troubleshooting Oracle Internet Directory Password Wallets ............................................... L-21
  Oracle Internet Directory Server Does Not Start ............................................................. L-21
  Password Not Synchronized ............................................................................................. L-22
List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2–1</td>
<td>Oracle Internet Directory Overview</td>
<td>2-6</td>
</tr>
<tr>
<td>3–1</td>
<td>A Typical Oracle Internet Directory Node</td>
<td>3-2</td>
</tr>
<tr>
<td>3–2</td>
<td>Oracle Directory Server Instance Architecture</td>
<td>3-4</td>
</tr>
<tr>
<td>3–3</td>
<td>A Directory Information Tree</td>
<td>3-7</td>
</tr>
<tr>
<td>3–4</td>
<td>Attributes of the Entry for Anne Smith</td>
<td>3-9</td>
</tr>
<tr>
<td>3–5</td>
<td>Correct and Incorrect Naming Contexts</td>
<td>3-14</td>
</tr>
<tr>
<td>3–6</td>
<td>A Replicated Directory</td>
<td>3-18</td>
</tr>
<tr>
<td>3–7</td>
<td>A Partitioned Directory</td>
<td>3-19</td>
</tr>
<tr>
<td>3–8</td>
<td>Using Knowledge References to Point to Naming Contexts</td>
<td>3-20</td>
</tr>
<tr>
<td>3–9</td>
<td>Oracle Identity Management Infrastructure and Other Components</td>
<td>3-24</td>
</tr>
<tr>
<td>3–10</td>
<td>Placement of Resource Access and Resource Type Information in the DIT</td>
<td>3-27</td>
</tr>
<tr>
<td>7–1</td>
<td>Directory Entry Hierarchy Showing Multiple Configuration Set Entries</td>
<td>7-2</td>
</tr>
<tr>
<td>7–2</td>
<td>Alias Entries Example</td>
<td>7-12</td>
</tr>
<tr>
<td>7–3</td>
<td>Resulting Tree when Creating the My_file.ldif</td>
<td>7-14</td>
</tr>
<tr>
<td>7–4</td>
<td>A Client Locating a Directory Server by Using DNS</td>
<td>7-18</td>
</tr>
<tr>
<td>10–1</td>
<td>Example of a Directory Information Tree</td>
<td>10-3</td>
</tr>
<tr>
<td>11–1</td>
<td>Location of Schema Components in Entries of Type subSchemaSubentry</td>
<td>11-2</td>
</tr>
<tr>
<td>14–1</td>
<td>Sample Audit Log in DSÉ</td>
<td>14-10</td>
</tr>
<tr>
<td>14–2</td>
<td>Architecture of Oracle Internet Directory Server Manageability</td>
<td>14-15</td>
</tr>
<tr>
<td>16–1</td>
<td>Indirect Authentication</td>
<td>16-6</td>
</tr>
<tr>
<td>18–1</td>
<td>Structural Access Item: Added Object Filter Tab Page</td>
<td>18-26</td>
</tr>
<tr>
<td>18–2</td>
<td>Structural Access Item: By Whom Tab Page</td>
<td>18-26</td>
</tr>
<tr>
<td>18–3</td>
<td>Example: Structural Access Item: Access Rights Tab Page</td>
<td>18-27</td>
</tr>
<tr>
<td>18–4</td>
<td>Content Access Item: By Whom Tab Page</td>
<td>18-28</td>
</tr>
<tr>
<td>18–5</td>
<td>Content Access Item: Attribute Tab Page</td>
<td>18-28</td>
</tr>
<tr>
<td>18–6</td>
<td>Content Access Item: Access Rights Tab Page</td>
<td>18-29</td>
</tr>
<tr>
<td>18–7</td>
<td>Content Access Item: By Whom Tab Page</td>
<td>18-30</td>
</tr>
<tr>
<td>18–8</td>
<td>Content Access Item: Attribute Tab Page</td>
<td>18-30</td>
</tr>
<tr>
<td>18–9</td>
<td>Content Access Item: Access Rights Tab Page</td>
<td>18-31</td>
</tr>
<tr>
<td>18–10</td>
<td>Content Access Item: By Whom Tab Page</td>
<td>18-32</td>
</tr>
<tr>
<td>18–11</td>
<td>Content Access Item: Attribute Tab Page</td>
<td>18-32</td>
</tr>
<tr>
<td>18–12</td>
<td>Content Access Item: Access Rights Tab Page</td>
<td>18-33</td>
</tr>
<tr>
<td>18–13</td>
<td>Content Access Item: By Whom Tab Page</td>
<td>18-34</td>
</tr>
<tr>
<td>18–14</td>
<td>Content Access Item: Attribute Tab Page</td>
<td>18-34</td>
</tr>
<tr>
<td>18–15</td>
<td>Content Access Item: Access Rights Tab Page</td>
<td>18-35</td>
</tr>
<tr>
<td>19–1</td>
<td>Location of Password Policy Entries</td>
<td>19-2</td>
</tr>
<tr>
<td>19–2</td>
<td>pwdPoliciesubentry Attributes Populated with DN of Password Policy</td>
<td>19-3</td>
</tr>
<tr>
<td>20–1</td>
<td>Location of the Password Verifier Profile Entry</td>
<td>20-5</td>
</tr>
<tr>
<td>20–2</td>
<td>Authentication Model</td>
<td>20-7</td>
</tr>
<tr>
<td>20–3</td>
<td>How Password Verification Works</td>
<td>20-9</td>
</tr>
<tr>
<td>21–1</td>
<td>Delegation Flow in an Oracle Application Server Environment</td>
<td>21-2</td>
</tr>
<tr>
<td>23–1</td>
<td>Planning the Directory Information Tree</td>
<td>23-2</td>
</tr>
<tr>
<td>23–2</td>
<td>Example of an Identity Management Realm</td>
<td>23-7</td>
</tr>
<tr>
<td>23–3</td>
<td>Enterprise Use Case: Single Identity Management Realm</td>
<td>23-8</td>
</tr>
<tr>
<td>23–4</td>
<td>Enterprise Use Case: Multiple Identity Management Realms</td>
<td>23-8</td>
</tr>
<tr>
<td>23–5</td>
<td>Hosted Deployment Use Case</td>
<td>23-9</td>
</tr>
<tr>
<td>23–6</td>
<td>Default Identity Management Realm</td>
<td>23-10</td>
</tr>
<tr>
<td>24–1</td>
<td>Usage Analysis of Current E-mail System</td>
<td>24-4</td>
</tr>
<tr>
<td>26–1</td>
<td>Example: Garbage Collection of Change Log Entries</td>
<td>26-4</td>
</tr>
<tr>
<td>26–2</td>
<td>Garbage Collection Entries in the DIT</td>
<td>26-5</td>
</tr>
<tr>
<td>27–1</td>
<td>Using an LDIF File and bulkload</td>
<td>27-3</td>
</tr>
<tr>
<td>27–2</td>
<td>Using dipassistant Directly</td>
<td>27-4</td>
</tr>
<tr>
<td>27–3</td>
<td>Using an LDIF File and dipassistant</td>
<td>27-5</td>
</tr>
<tr>
<td>27–4</td>
<td>Using dipassistant, bulkload, and LDIF Files</td>
<td>27-5</td>
</tr>
</tbody>
</table>
List of Tables

1–1 Object Classes and Attributes .................................................................................................. 1–1
1–2 Replication ................................................................................................................................. 1–2
1–3 Security, Password Policies, and User Accounts................................................................. 1–3
1–4 Realms ........................................................................................................................................... 1–3
1–5 Server Processes, Instances, and Configuration Set Entries ........................................... 1–4
1–6 System Operational Attributes ............................................................................................... 1–5
1–7 Naming Contexts ....................................................................................................................... 1–5
1–8 Binds, Connections, Aliases, and Directory Discovery ....................................................... 1–5
1–9 Referential Integrity ................................................................................................................. 1–6
1–10 Entries ........................................................................................................................................ 1–6
1–11 Groups ...................................................................................................................................... 1–8
1–12 Logging, Auditing, and Monitoring ....................................................................................... 1–8
1–13 Tuning ....................................................................................................................................... 1–8
1–14 Garbage Collection ................................................................................................................. 1–9
1–15 Server Chaining and Data Migration ................................................................................... 1–9
1–16 Plug-ins ................................................................................................................................... 1–10
2–1 Comparison of Online Directories and Relational Databases ........................................ 2–2
3–1 Components of an Oracle Internet Directory Node .............................................................. 3–3
3–2 Attributes Created with Each New Entry ............................................................................. 3–10
3–3 Common LDAP Attributes ........................................................................................................ 3–10
4–1 Tasks to Reset the Default Security Configuration ............................................................ 4–2
5–1 Operating System-Specific Instructions for Starting Oracle Directory Manager ............ 5–2
5–2 Oracle Directory Manager Menu Bar ..................................................................................... 5–4
5–3 Oracle Directory Manager Toolbar ....................................................................................... 5–5
5–4 Task Areas in Oracle Directory Manager ............................................................................. 5–7
5–5 Tools for Starting, Stopping, and Monitoring Oracle Internet Directory Servers ............ 5–9
5–6 Tools for Managing Entries ..................................................................................................... 5–10
5–7 Command-Line Tools for Performing Bulk Operations ................................................... 5–11
5–8 Command-Line Tools for Managing Replication ............................................................... 5–12
6–1 Process Control Items in the ODS_PROCESS Table .............................................................. 6–6
7–1 Definitions of Super User, Guest User, and Proxy User ..................................................... 7–9
7–2 Names, Passwords, and Attributes for Super, Guest, and Proxy Users ........................... 7–10
7–3 Orclanonymousbindflag Value and Directory Server Behavior ........................................ 7–10
7–4 Entry Alias Dereferencing Messages ................................................................................... 7–16
7–5 Arguments in a Service Location Record (SRV) ................................................................. 7–19
8–1 Command-Line Tools for Managing Entries ..................................................................... 8–7
10–1 Attribute Uniqueness Constraint Entry ............................................................................... 10–2
11–1 Content Rule Parameters ....................................................................................................... 11–20
11–2 Attribute Aliases Used in Examples ................................................................................... 11–23
13–1 orclDynamicGroup Attributes for “Connect By” Assertions ......................................... 13–4
13–2 Static and Dynamic Group Considerations ........................................................................ 13–6
14–1 Log File Locations ................................................................................................................ 14–1
14–2 Fields in Trace Messages ...................................................................................................... 14–5
14–3 Debug Logging Levels ........................................................................................................ 14–6
14–4 Debug Dimension Values for LDAP Operations ............................................................ 14–6
14–5 Attributes of the orclAuditoc Object Class ................................................................... 14–9
14–6 Auditable Events .................................................................................................................. 14–10
14–7 Audit Mask Levels ............................................................................................................... 14–11
14–8 Example: Setting the Audit Level ...................................................................................... 14–12
14–9 Components of Oracle Internet Directory Server Manageability ................................... 14–15
14–10 Values of the DSA Configuration Attribute orcloptracklevel ........................................ 14–18
14–11 Metrics Recorded by Each orcloptracklevel Value ....................................................... 14–18
14–12 Critical Event Levels ......................................................................................................... 14–19
Fields in the Start a New LDAP Server Instance Window of the Application Server Control Console ................................................. 14-21
Fields in the Restart an LDAP Server Instance Window of the Application Server Control Console .......................................... 14-21
SSL Cipher Suites Supported in Oracle Internet Directory ................................................ 17-1
Sample Security Groups ........................................................................................................ 18-5
Types of Access ........................................................................................................................ 18-9
LDAP Operations and Access Needed to Perform Each One ................................................. 18-10
Attribute States During ACL Evaluation ............................................................................ 18-10
DNs Used in Example ............................................................................................................ 18-39
Password Policy Attributes .................................................................................................. 19-5
Tasks and Tools for Managing Password Polices ................................................................. 19-8
Attributes for Storing Password Verifiers in User Entries .................................................. 20-5
Default Privileges Granted to Everyone and to Each User .................................................. 21-3
Privileges for Administering the Oracle Technology Stack ............................................... 21-4
Characteristics of the Subscriber DAS Create User Group .................................................. 21-5
Characteristics of the Subscriber DAS Edit User Group ........................................................ 21-6
Characteristics of the DAS Delete User Group .................................................................... 21-6
Characteristics of the User Privilege Assignment Group ..................................................... 21-6
Characteristics of the Group Creation Group ..................................................................... 21-7
Characteristics of the Group Edit Group ............................................................................. 21-7
Characteristics of the Group Delete Group ......................................................................... 21-8
Characteristics of the Group Privilege Assignment Group .................................................. 21-8
Characteristics of the Oracle Application Server Administrators Group ............................ 21-9
Characteristics of the User Management Application Administrators Group ..................... 21-10
Characteristics of the Trusted Application Administrators Group ...................................... 21-10
Characteristics of the User Security Administrators Group ................................................. 21-12
Characteristics of the Authentication Services Group ......................................................... 21-12
Characteristics of the Verifier Services Group .................................................................... 21-12
Characteristics of the User Proxy Privilege Group ............................................................... 21-13
Characteristics of the Oracle Context Administrators Group ............................................. 21-13
Characteristics of the Common User Attributes Group ...................................................... 21-14
Characteristics of the Common Group Attributes Group .................................................... 21-14
Characteristics of the Service Registry Viewers Group ....................................................... 21-14
Characteristics of the Common Group Attributes Group .................................................... 21-14
CPU Power for Various Deployment Scenarios .................................................................... 22-7
Approximate Disk Space Requirements for Variously Sized DITs ...................................... 22-7
Estimates of the Memory Requirements for Various DIT Sizes ......................................... 22-7
Oracle Identity Management Objects .................................................................................... 23-9
Customizing the Default Identity Management Realm ....................................................... 23-12
Capacity Planning Terminology ........................................................................................... 24-2
Assumptions about Entry Types and Their Sizes ................................................................. 24-2
Overall Count of Entries ...................................................................................................... 24-3
Directory Lookups in a Single Day ....................................................................................... 24-3
Working Hour Loads ............................................................................................................. 24-4
Disk Space Requirements .................................................................................................... 24-5
Tablespaces Used to Store Oracle Internet Directory Data ................................................. 24-6
Variables Used for Size Calculation .................................................................................... 24-6
Size of Individual Tablespaces ............................................................................................ 24-7
Values for Variables Used for Sizing Calculations ............................................................... 24-8
Tablespace Sizes .................................................................................................................. 24-9
Minimum Memory Requirements for Different Directory Configurations .......................... 24-10
Maximum Possible Throughput for Two Types of Operations ........................................... 24-10
Rough Estimates of CPU Requirements .............................................................................. 24-12
Suggested Values for ORCLSERVERPROCS and ORCLMAXCC Parameters ................. 25-3
<table>
<thead>
<tr>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>25-2</td>
</tr>
<tr>
<td>27-1</td>
</tr>
<tr>
<td>27-2</td>
</tr>
<tr>
<td>28-1</td>
</tr>
<tr>
<td>28-2</td>
</tr>
<tr>
<td>28-3</td>
</tr>
<tr>
<td>29-1</td>
</tr>
<tr>
<td>29-2</td>
</tr>
<tr>
<td>29-3</td>
</tr>
<tr>
<td>29-4</td>
</tr>
<tr>
<td>29-5</td>
</tr>
<tr>
<td>29-6</td>
</tr>
<tr>
<td>29-7</td>
</tr>
<tr>
<td>29-8</td>
</tr>
<tr>
<td>29-9</td>
</tr>
<tr>
<td>29-10</td>
</tr>
<tr>
<td>29-11</td>
</tr>
<tr>
<td>30-1</td>
</tr>
<tr>
<td>30-2</td>
</tr>
<tr>
<td>30-3</td>
</tr>
<tr>
<td>A-1</td>
</tr>
<tr>
<td>A-2</td>
</tr>
<tr>
<td>A-3</td>
</tr>
<tr>
<td>A-4</td>
</tr>
<tr>
<td>A-5</td>
</tr>
<tr>
<td>A-6</td>
</tr>
<tr>
<td>A-7</td>
</tr>
<tr>
<td>A-8</td>
</tr>
<tr>
<td>A-9</td>
</tr>
<tr>
<td>A-10</td>
</tr>
<tr>
<td>A-11</td>
</tr>
<tr>
<td>A-12</td>
</tr>
<tr>
<td>A-13</td>
</tr>
<tr>
<td>A-14</td>
</tr>
<tr>
<td>A-15</td>
</tr>
<tr>
<td>A-16</td>
</tr>
<tr>
<td>A-17</td>
</tr>
<tr>
<td>A-18</td>
</tr>
<tr>
<td>A-19</td>
</tr>
<tr>
<td>A-20</td>
</tr>
<tr>
<td>A-21</td>
</tr>
<tr>
<td>A-22</td>
</tr>
<tr>
<td>A-23</td>
</tr>
<tr>
<td>A-24</td>
</tr>
<tr>
<td>A-25</td>
</tr>
<tr>
<td>A-26</td>
</tr>
<tr>
<td>A-27</td>
</tr>
<tr>
<td>A-28</td>
</tr>
<tr>
<td>A-29</td>
</tr>
<tr>
<td>A-30</td>
</tr>
<tr>
<td>A-31</td>
</tr>
<tr>
<td>A-32</td>
</tr>
<tr>
<td>A-33</td>
</tr>
<tr>
<td>A-34</td>
</tr>
<tr>
<td>A-35</td>
</tr>
</tbody>
</table>
Preface

Oracle Internet Directory Administrator’s Guide describes the features, architecture, and administration of Oracle Internet Directory. For information about installation, see the installation documentation for your operating system.

This Preface contains these topics:

■ Audience
■ Documentation Accessibility
■ Related Documents
■ Conventions

Audience

Oracle Internet Directory Administrator’s 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 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 Documents**
For more information, see:

- Online help available through Oracle Directory Manager, the Oracle Internet Directory Single Sign-On Console, and Oracle Enterprise Manager 10g
- The Oracle Application Server and Oracle Database documentation sets, especially:
  - Oracle Identity Management Infrastructure Administrator’s Guide
  - Oracle Identity Management Integration Guide
  - Oracle Identity Management Guide to Delegated Administration
  - Oracle Identity Management Application Developer’s Guide
  - Oracle Application Server Single Sign-On Administrator’s Guide
  - Oracle Application Server Certificate Authority Administrator’s Guide
  - Oracle Identity Management User Reference
  - Oracle Application Server High Availability Guide
  - Oracle Application Server Administrator’s Guide
  - Oracle Database Administrator’s Guide
  - Oracle Database Net Services Administrator’s Guide
  - Oracle Database Oracle Clusterware and Oracle Real Application Clusters Administration and Deployment Guide
  - Oracle Database Advanced Replication
  - Oracle Advanced Security Administrator’s Guide

For additional information, see:

- Internet Assigned Numbers Authority home page, [http://www.iana.org](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 LDAP 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:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface</strong></td>
<td>Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.</td>
</tr>
<tr>
<td>italic</td>
<td>Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.</td>
</tr>
<tr>
<td>monospace</td>
<td>Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.</td>
</tr>
</tbody>
</table>
What's New in Oracle Internet Directory?

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 Internet Directory 10g Release 2 (10.1.2)
- New Features Introduced with Oracle Internet Directory 10g (9.0.4)
- About Oracle Internet Directory Release 9.2
- New Features Introduced with Oracle Internet Directory Release 9.0.2
- New Features Introduced with Oracle Internet Directory Release 3.0.1
- New Features Introduced with Oracle Internet Directory Release 2.1.1

New Features Introduced with Oracle Internet Directory 10g (10.1.4.0.1)

- **Links to Procedural Information**: This document contains a table of links to important tasks. See Chapter 1, "Links to Common Tasks".

- **Identity Management Grid Control Plug-in**: This new interface enables you to monitor and manage Oracle Internet Directory, Oracle Application Server Single Sign-On, Oracle Delegated Administration Services, and Oracle Directory Integration Platform, using the features of the Oracle Enterprise Manager 10g Grid Control Console.

  **See Also**: The Identity Management Grid Control Plug-in chapter in *Oracle Identity Management Infrastructure Administrator’s Guide*

- **Improved Bulk Tools**: The following bulk tools have been converted into C executables:
  - bulkload
  - bulkmodify
  - bulkdelete
  - catalog
  - ldifwrite

Examples and descriptions in this document and in *Oracle Identity Management User Reference* have been updated to reflect the new features of these tools.
Application-Specific Schema Containers: A product that adds schema to Oracle Internet Directory can have its own subSchemaSubentry under cn=subSchemaSubentry.

See Also: "About the Directory Schema" in Chapter 11, "Directory Schema Administration"

Support for Attribute Aliases: You can create user-friendly aliases for attribute names.

See Also: "Attribute Aliases In the Directory" in Chapter 11, "Directory Schema Administration"

Caching of Dynamic Groups: Dynamic group members are computed when the dynamic group is added, and the member list is kept consistent when the dynamic group is later modified.

See Also: "Dynamic Groups" in Chapter 13, "Dynamic and Static Groups in Oracle Internet Directory"

Optimizing Searches for Large Group Entries: There is an additional technique for optimizing searches by increasing the size of the entry cache instead of disabling the entry cache.

See Also: "Optimizing Searches for Large Group Entries" in Chapter 25, "Tuning Considerations for the Directory"

Referential Integrity: If you enable Referential Integrity, whenever you update an entry in the directory, the server also updates other entries that refer to that entry.

See Also: Chapter 12, "Referential Integrity"

New Monitoring Capabilities for Server Manageability: You can enable additional health statistics, user statistics, and security events tracking.

See Also: "Monitoring Oracle Internet Directory Servers" in Chapter 14, "Logging, Auditing, and Monitoring the Directory"

New Password Policy Features: You can apply a password policy to any subtree, or even a single entry. There are also more password policy attributes to choose from.

See Also: Chapter 19, "Password Policies in Oracle Internet Directory"

Server Chaining: This feature enables you to map entries that reside in third party LDAP directories to part of the directory tree and access them through Oracle Internet Directory, without synchronization or data migration.

See Also: Chapter 28, "Server Chaining"
■ **Paging and Sorting of LDAP Search Results**: The ldapsearch command now has a `-T` option for sorting and a `-j` option for paging.

**See Also:**

The ldapsearch command-line reference in *Oracle Identity Management User Reference*

The chapter on extensions to the LDAP protocol in *Oracle Identity Management Application Developer’s Guide*

■ **New Replication Features**: Oracle Internet Directory Replication has been enhanced with the following features:

– **Two-way LDAP-Based Replication**: This feature enables you to deploy fan-out replication groups where replication flows in both directions and updates at any node are replicated to the whole group.

– **Replication Failover**: Failover of LDAP replicas from one supplier to another is supported, with administrator intervention.

– **Oracle Internet Directory Comparison and Reconciliation Tool**: A new oidcmprec command, with improved functionality, replaces the old oidreconcile tool.

**See Also:**

Chapter 29, "Oracle Internet Directory Replication Concepts"

Chapter 30, "Oracle Internet Directory Replication Installation and Configuration"

Chapter 31, "Oracle Internet Directory Replication Monitoring and Management"

The oidcmprec command-line tool reference in *Oracle Identity Management User Reference*

■ **Java Server Plug-ins**: The Oracle Internet Directory Plug-in Framework now supports plug-ins written in Java as well as in PL/SQL.

**See Also**: Chapter 32, "Oracle Internet Directory Server Plug-in Framework"
New Features Introduced with Oracle Internet Directory 10g Release 2 (10.1.2)

Note:
The following chapters have been moved to Oracle Application Server High Availability Guide:

- "High Availability And Failover Considerations"
- "Oracle Application Server Cluster (Identity Management) Configurations"
- "Oracle Application Server Cold Failover Cluster (Identity Management)"
- "The Directory in an Oracle Real Application Clusters Environment"

The following appendixes have been rewritten as chapters in Oracle Identity Management User Reference:

- "Syntax for LDIF and Command-Line Tools"
- "Oracle Internet Directory Schema Elements"

- Improved integration with other components: New features provide better integration with components such as Oracle Collaboration Suite. These features include service-to-service authentication, the service registry, and verifier generation using dynamic parameters.

  See Also:
  - "The Service Registry and Service to Service Authentication" on page 3-21
  - “Verifier Generation Using Dynamic Parameters” on page 20-10

- Support for Certificate Matching Rule: External authentication using certificates can now take either of two forms: an exact match, in which the subject DN of the client certificate is used to authenticate the user, or a certificate hash, in which the client certificate is hashed and is then compared with a certificate hash stored in the directory.

  See Also:  "Direct Authentication" on page 16-4

- Ease of deployment for Replication: Replication is now much easier to install, configure, and manage.

  See Also:
  
  Chapter 30, "Oracle Internet Directory Replication Installation and Configuration"
  
  Oracle Application Server Installation Guide

- Ease of deployment for Clusters: Cluster configurations are now much easier to install, configure, and manage.
Enforcing access control for Oracle Internet Directory super user: The super user is now subject to access control policies like any other user. New ACL keywords allow you to restrict super user access through privileged groups.

See Also: Chapter 18, "Directory Access Control"

Oracle Internet Directory Server Diagnostic Tool: The OID Diagnostic Tool collects diagnostic information that helps triage issues reported on Oracle Internet Directory.

See Also: The oiddiag command-line tool reference in Oracle Identity Management User Reference

New Features Introduced with Oracle Internet Directory 10g (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 Platform and plug-ins.

See Also: The chapter on integration with Microsoft Windows in the Oracle Identity Management Integration Guide

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:
- Chapter 34, "Setting Up the Customized External Authentication Plug-in"
- The chapter on considerations for integrating with third-party connected directories in Oracle Identity Management Integration Guide

Dynamic groups: You can create and use dynamic groups whose membership, rather than being maintained in a list, is computed on the fly, based on assertions that you specify.

See Also: Chapter 13, "Dynamic and Static Groups in Oracle Internet Directory"

Query optimization: In searches, some attributes have very different response times depending on their values. You can uniform the response times of search operations for such attributes to enhance performance.

See Also: "Optimizing Searches" on page 25-9
Garbage collection framework: A garbage collector is a background database process that removes obsolete data from the directory. The Oracle Internet Directory garbage collection framework provides a default set of garbage collectors, and enables you to modify them.

See Also: Chapter 26, "Garbage Collection in Oracle Internet Directory"

Simple Authentication Security Layer (SASL) support: Oracle Internet Directory supports the use of SASL, a method for adding authentication support to connection-based protocols. To use it, a protocol includes a command for identifying and authenticating a user to a server and for optionally negotiating protection of subsequent protocol interactions. If its use is negotiated, a security layer is inserted between the protocol and the connection.

See Also: "Authentication in Oracle Internet Directory" on page 16-4

Logging enhancements: This release of Oracle Internet Directory provides the following enhancements to logging and tracing:

- Object-based tracing for operations associated with thread and connection identifiers. This facilitates non-interleaved and coherent logging for each LDAP operation in a multithreaded environment.
- Selective tracing for chosen operations by using the operation dimension.
- Structured, meaningful trace messages with additional information including thread identifier and criticality.

See Also: Chapter 14, "Logging, Auditing, and Monitoring the Directory"

OID Migration Tool (ldifmigrator) enhancements: You can use this tool to reconcile data with that in an existing directory, and to directly load data into Oracle Internet Directory.

See Also:

- "Migrating User Data from Application-Specific Repositories" on page 27-8
- The ldifmigrator command-line tool reference in Oracle Identity Management User Reference

Client side referral caching: This new feature enables clients to cache referral information and use it to speed up referral processing.

See Also:

- "Client-Side Referral Caching" on page 8-11
- The material on the ldap_set_option and the ldap_get_option in Oracle Identity Management Application Developer's Guide

Fan-out and partial replication support: Oracle Internet Directory now supports:

- Partial replication: that is, propagation of one or more naming contexts, rather than the entire DIT, to another node
– Fan-out replication, in which a consumer, having received changes from a supplier, can then replicate those changes to one or more other consumers. Fan-out replication can be either full or partial.

**See Also:**
- Chapter 29, "Oracle Internet Directory Replication Concepts"
- Chapter 30, "Oracle Internet Directory Replication Installation and Configuration"

**Password policy enhancements:** New password policy capabilities in Oracle Internet Directory include:
- Password history
- Unlocking of accounts
- Forced password change upon first login
- Self-resetting of password in case of account lockout or forgotten passwords
- Super user account lockout requiring reset.
- IP-based account lockout
- Password policy enablement or disablement by using a single attribute in the password policy entry

**See Also:** Chapter 19, "Password Policies in Oracle Internet Directory"

**Security credential storage enhancements:** New security credential storage capabilities in Oracle Internet Directory include:
- Generation of O3logon verifier for enterprise users
- Generation of a default set of verifiers for application bootstrapping
- Generation of SASL/MD5 verifiers for directory authentication

**See Also:** Chapter 20, "Directory Storage of Password Verifiers"

**Replication Environment Management Tool:** This tool ensures that Oracle Advanced Replication is properly configured for directory replication. In the event of a directory replication failure, this tool looks for common problems and seeks to rectify them. If it cannot solve the problem, then it gives you a report of the nature of the problem and points you to a possible solution.

**See Also:** The remtool command-line tool reference in Oracle Identity Management User Reference

**Server discovery by using DNS:** This feature enables the location of a directory server in a distributed environment to be discovered dynamically by using the domain name system (DNS). Rather than storing server location information statically in an ldap.ora file on the client, that information is stored and managed in a central domain name server. The client, at request processing time, retrieves this information from the domain name server.

**Bulkload tool enhancements:** You can now use bulkload to add a large volume of entries to a non-empty directory. For example, you can add one million entries to a directory that has one million entries already. You can also incrementally add a
medium-size number of entries to a large directory. For example, you can add 50,000 entries at a time to a directory that has five million entries already.

**See Also:** The `bulkload` command-line tool reference in *Oracle Identity Management User Reference*

- **Oracle Application Server Cluster (Identity Management) directory server configuration support:** This configuration provides high availability of a directory server by running multiple directory server instances on different hardware nodes. The directory servers are connected to the same underlying data store, which is an Oracle Database.

  **See Also:** "Oracle Application Server Cluster (Identity Management) Configurations" in *Oracle Application Server High Availability Guide*

- **Two-way provisioning between Oracle Internet Directory and other application directories:** The Oracle Directory Provisioning Integration Service can send notification of provisioning events bidirectionally between Oracle Internet Directory and other applications.

  **See Also:** The chapter on provisioning service concepts in the *Oracle Identity Management Integration Guide*

- **Integration of provisioning data with the Oracle E-Business Suite:** You can synchronize user accounts and other user information from the Oracle E-Business Suite to Oracle Internet Directory by using the Oracle Directory Provisioning Integration Service.

  **See Also:** The chapter on integration with the Oracle E-Business Suite in the *Oracle Identity Management Integration Guide*

- **Installation of Oracle Internet Directory on Oracle Real Application Clusters:** You can install Oracle Internet Directory on Oracle Real Application Clusters. When you do this, both the software and schema for Oracle Internet Directory are installed on the primary node, while only the software is installed on the secondary nodes.

  **See Also:** The installation documentation for this release of Oracle Internet Directory

- **Oracle Directory Manager enhancements:** Oracle Directory Manager now enables you to manage the following:
  - Attribute uniqueness
  - Plug-ins
  - Garbage collection
  - Change logs
  - Replication
  - Query optimization
  - Debug logging to a finer degree than previously
  - Enhancement of ACLs
Oracle Internet Directory Self-Service Console enhancements: Oracle Internet Directory Self-Service Console, a graphical administrative tool built with Oracle Delegated Administration Services units, enables you to manage the following:

- Realms
- Services
- Accounts
- Password resetting

Oracle Internet Directory Self-Service Console also enables you to view your organization chart, and users to edit their own profiles.

See Also: The chapter about the Oracle Internet Directory Self-Service Console in Oracle Identity Management Guide to Delegated Administration

Upgrade procedures

See Also: Oracle Application Server Upgrade and Compatibility Guide for information about upgrading from an earlier version of Oracle Internet Directory

About Oracle Internet Directory Release 9.2

This section describes an important new feature employing the capabilities of Oracle Internet Directory. It also explains changes in Oracle Internet Directory since Release 9.0.2.

User Migration Utility for bulk-migrating database users to Oracle Internet Directory: This utility, released with Oracle Advanced Security Release 2 (9.2), enables you to migrate users from a local or external database to Oracle Internet Directory. Use it to store and centrally manage thousands of users in Oracle Internet Directory.

See Also: The chapter about migrating local or external users to enterprise users in Oracle Advanced Security Administrator’s Guide

Note:

- Beginning with Oracle Internet Directory Release 9.2, the Oracle Delegated Administration Services and tools built on it are components of Oracle Application Server and not the Oracle Database. To ensure that you have the self-management tools for administering Web and Oracle Application Server applications, and that those tools are well-integrated with your middle-tier environment, Oracle recommends that you use the version of Oracle Internet Directory that is included with the Oracle Application Server. To develop and deploy tools based on the Oracle Delegated Administration Services, Oracle Corporation recommends that you use the Java and security infrastructure of Oracle Application Server.

- Oracle Internet Directory Release 9.2 does not include Enterprise Manager integration for performing system diagnostics on Oracle Internet Directory instances.
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.

- **Server-side entry caching**: This feature reduces directory query latency for LDAP clients. By configuring a server-side entry cache based on naming context, identity of client, or other available parameters, Oracle Internet Directory ensures that previously retrieved entries and their attributes are stored in shared memory, and are thus available to subsequent data requestors. Queries that conform to the configured parameters then need only retrieve a small subset of data—internal globally unique identifiers (GUIDs)—for filter-matching entries from the directory. These returned GUIDs are then used as a fast lookup mechanism into the cached entry and attribute data, which is then returned to the client.

  **See Also**: "Entry Caching" on page 25-8

- **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 Directory Provisioning Integration Service and Oracle Directory Synchronization Service are built upon Oracle Directory Integration Platform (introduced with Oracle Internet Directory v2.1.1.1 in the Oracle8i Release 3 time frame).

  - **Oracle Directory Provisioning Integration 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 Directory Provisioning Integration Service ensures that subscribing applications or business entities are alerted to updates in Oracle Internet Directory for keeping local repositories in synch. 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.

  **See Also**: The chapter on concepts and components in the Oracle Identity Management Integration Guide

- **Enterprise password policy management enhancements**: You can now construct password policies to ensure:

  - Expiration dates
  - Grace periods
  - Minimum password lengths
  - Approved password syntaxes and retry limits
  - Lockout of those attempting to gain illicit access to the directory service after a certain number of failed attempts
You can now use salted SHA as a hashing algorithm. This means that you can now select from these available hashing algorithms:

- **MD4**: A one-way hash function that produces a 128-bit hash
- **MD5**: An improved, and more complex, version of MD4
- **SHA**: Secure Hash Algorithm, which produces a 160-bit hash, longer than MD5. The algorithm is slightly slower than MD5, but the larger message digest makes it more secure against brute-force collision and inversion attacks.
- You can also use salted SHA. A salt is a random number added to and stored with the hash value. It prevents pre-computed dictionary attacks by making it extremely expensive to recover the value that was originally hashed.
- **UNIX Crypt**: The UNIX encryption algorithm
- **No Hashing**

**See Also:**

- "Protection of User Passwords for Directory Authentication" on page 16-7 for a conceptual discussion
- Chapter 19, "Password Policies in Oracle Internet Directory" for instructions on setting password hashing

**Attribute uniqueness**: In the prior Oracle Internet Directory architecture, the only way to enforce attribute uniqueness was to make an attribute a part of your DN. This worked well with the user identifier (if used as the RDN), but it was not always appropriate and easy to configure. Within a level of a branch of the tree, it was guaranteed to be unique. For example, if your DN was `uid=dlin,ou=people, o=oracle`, then the RDN `dlin` would be unique directly under `ou=people, o=oracle`. However, you could have the same user identifier in another branch: for example, `uid=dlin, ou=others, o=oracle`. In short, attribute uniqueness was guaranteed only under a given branch, and only within one level.

Attributes other than `dn` can be used as unique keys of applications synchronizing with Oracle Internet Directory. The ability of Oracle Internet Directory to enforce attribute uniqueness enables all applications to have their own notions of "user," and to synchronize their user base with a user repository stored in an enterprise Oracle Internet Directory server.

**See Also**: Chapter 10, "Attribute Uniqueness in the Directory"

**Multiple password verifier support**: Oracle Internet Directory can now store passwords for multiple applications and protocols. For example, four-digit Personal Identification Numbers (PINs) for voicemail can sit alongside longer alphanumeric single sign-on passwords and X.509 v3 digital certificates for the same user. This new feature gives the application developer far greater flexibility for directory-enabling their product stack.

**See Also**: Chapter 20, "Directory Storage of Password Verifiers"

**Expanded proxy user capabilities**: This new feature enables a developer to exploit the power of the middle tier more effectively. Users no longer need to establish independent, unrelated sessions with the directory. If a middle-tier from Oracle Application Server or elsewhere invokes the proxy user bind method on behalf of numerous clients in succession, then Oracle Internet Directory respects each
client’s credential and privileges respectively, even though the agent doing the actual binding remains unchanged throughout.

See Also:
- Chapter 16, "Directory Security Concepts"
- "Managing Super Users, Guest Users, and Proxy Users" on page 7-8

Integration with Oracle Application Server components: Through the Oracle Directory Provisioning Integration Service, Oracle Internet Directory Release 9.0.2 serves as a central component of the Oracle Application Server. Every component of Oracle Application Server now uses Oracle Internet Directory for storing common cross-component metadata, such as valid user identifiers and their passwords.

See Also: Chapter 23, "Deployment of Oracle Identity Management Realms"

Enterprise Manager integration: You can start, stop, and monitor Oracle Internet Directory instances by using the standard, newly-enhanced Enterprise Manager console. You can perform system diagnostics on running Oracle Internet Directory instances, and generate performance graphs to determine ongoing performance and peak load times.

See Also: Monitoring Oracle Internet Directory Servers on page 14-13

Oracle Directory Manager enhancements: Oracle Internet Directory’s standalone, 100% Java administration console, Oracle Directory Manager, has now evolved in many ways. You can use it to:
- Configure realms
- Construct password policies
- Configure Oracle Directory Synchronization Service and Oracle Internet Directory connectors and agents

In general, any directory-specific configuration or maintenance task not available at the high-level Oracle Enterprise Manager GUI can now be done through Oracle Directory Manager as well as command-line interfaces supplied with Oracle Internet Directory.

See Also: Chapter 5, "Directory Administration and Monitoring Tools"

Server-side plug-in framework: This new feature enables directory applications to roll out advanced capabilities such as referential integrity/cascading deletions of LDAP objects, external authentication of directory clients, brokered access, and synchronization with external relational tables. The plug-ins are executable before or after an LDAP command takes place, without the traditional risks of such technologies.

See Also: Chapter 32, "Oracle Internet Directory Server Plug-in Framework"
- **Entry alias dereferencing**: The LDAP v3 standard requires that all entries in a directory have globally unique identifiers known as distinguished names. These are typically fairly long and cumbersome to use, so Oracle Internet Directory provides this new feature to automatically dereference IETF-standard alias objects used to point to a fully-qualified LDAP distinguished name. For example, "DavesServer1" can be used as an entry alias or pointer to the actual directory entry named `dc=server1, dc=us, dc=oracle, dc=com`. Oracle Internet Directory stores, parses, and chases all alias references for complete client-side transparency.

  **See Also**:  "Dereferencing Alias Entries" on page 7-12

- **Delegated Administration Service**

  The Oracle Delegated Administration Services is a set of individual, pre-defined services—called Oracle Delegated Administration Services units—for performing directory operations on behalf of a user. It makes it easier to develop and deploy administration solutions for both Oracle directory-enabled applications and other directory-enabled applications that use Oracle Internet Directory.

  Administrators can now use the Oracle Delegated Administration Services and its accompanying console to:

  - Create other regional or departmental administrators
  - Grant them specific, delegated permissions to administer users for a particular region or department

  The Oracle Internet Directory Self-Service Console, a new component of the Oracle Delegated Administration Services, enables you to flexibly administer applications, realms, and end users either from a central team or through decentralization and delegation. It provides:

  - A unified resource for directory administrators, directory service subscribers, and end users
  - A view of an authorized end user’s personalized preferences and the ability to update their Oracle Application Server Single Sign-On password
  - An intuitive user interface for searching for people and other directory-based resource information within Oracle Internet Directory.

  You can use the Oracle Internet Directory Self-Service Console to configure the object classes, user groups, permissions, and other elements of directory information metadata stored in Oracle Internet Directory.

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

- **Upgrade procedures**

  These procedures enable you to upgrade from Oracle Internet Directory release 2.1.1. and release 3.0.1.

**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.
Failover in cluster configurations

This new feature enables you to increase high availability by using logical hosts—as opposed to physical hosts—in clustered environments.

See Also: "Oracle Application Server Cold Failover Cluster (Identity Management)" in Oracle Application Server High Availability Guide

Failover in an Oracle Real Application Clusters environment

Oracle Real Application Clusters is a computing environment that harnesses the processing power of multiple, interconnected computers. Along with a collection of hardware, called a cluster, it unites the processing power of each component to become a single, robust computing environment. A cluster comprises two or more computers, also called nodes.

You can run Oracle Internet Directory in an Oracle Real Application Clusters system.

See Also: "The Directory in an Oracle Real Application Clusters Environment" in Oracle Application Server High Availability Guide

Support for logical hosts: Oracle Internet Directory Release 3.0.1 enables you to increase high availability by using logical hosts – as opposed to physical hosts – in clustered environments. A logical host consists of 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 services the host name and IP address of the logical host.

In this paradigm, the directory server binds to the logical host, rather than the physical host. It maintains this connection even if the logical host fails over to a new physical host.

A client connects to the directory server by using the logical host name and address of the server. If the logical host fails over to a new physical host, then that failover is transparent to the client.

See Also: "Oracle Application Server Cold Failover Cluster (Identity Management)" in Oracle Application Server High Availability Guide

Capability to run multiple Oracle Internet Directory instances on the same host

This new feature enables you to run more than one installation of Oracle Internet Directory on a single host. You can then replicate between them or use this new feature as part of a failover strategy.

See Also: "Multiple installations of Oracle Internet Directory on one host" on page 22-5

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.

See Also: Oracle Identity Management Integration Guide
Password policy management

Password policy management enables you to establish and enforce rules for how passwords are used.

See Also:
- "Protection of User Passwords for Directory Authentication" for a conceptual discussion
- Chapter 19, "Password Policies in Oracle Internet Directory"

Performance and scalability enhancements

Upgrade procedures

These procedures enable you to upgrade from Oracle Internet Directory release 2.1.1.

UTF8 restriction removed

The Oracle directory server and database tools are no longer restricted to run on a UTF8 database. However, there may be data loss during add, delete, modify, or modifydn operations if the character sets of the data contained in the client request and the directory server database repository are different and the client data cannot be mapped to the database character set. If the database underlying the Oracle directory server is neither AL32UTF8 nor UTF8, then be sure that all characters in the client character set are included in the database character set, with the same or different character codes.

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.

Attribute options, including language codes

Attribute options enable you to specify how the value for an attribute is made available in a search or a compare operation. For example, suppose that an employee has two addresses, one in London, the other in New York. Options for that employee’s address attribute could allow you to store both addresses. Users could then search for either address.

Attribute options can include language codes. For example, options for John Doe’s givenName attribute could enable you to store his given name in both French and Japanese. A user could then search for the name in either language.

See Also:
- "Attribute Options" on page 3-11 for a conceptual discussion
- "Managing Entries with Attribute Options by Using Oracle Directory Manager" on page 8-6
- "Managing Entries with Attribute Options by Using Command-Line Tools" on page 8-9

Change log purging enhancements

These enhancements enable you to specify the type of change log purging to use: change number-based or time-based.
Enhanced support for these operational attributes: creatorsName, createTimestamp, modifiersName, and modifyTimestamp

This enhanced support enables you to use one or more of these attributes in searches.

See Also:

- "Kinds of Attribute Information" on page 3-9 for a conceptual discussion
- "Change Log Purging in Multimaster Replication" on page 26-5 for a conceptual discussion
- "Viewing and Modifying Parameters for a Particular Replica Node by Using Oracle Directory Manager" on page 31-4

Migration from other LDAP-compliant directories

This new feature enables you to migrate data from other LDAP v3-compatible directories into Oracle Internet Directory.

See Also: Appendix 27, "Migration of Data from Other Data Repositories"

Object class explosion

Object class explosion enables you to add or perform an operation on an entry without specifying the entire hierarchy of superclasses associated with that entry.

See Also: "Guidelines for Adding Object Classes" on page 11-3 for an explanation of how to use this feature when adding object classes

OID Database Statistics Collection tool

This tool assists in capacity planning. It helps you analyze the various database schema objects so that you can estimate the statistics.

See Also: The oidstats.sql command-line tool reference in Oracle Identity Management User Reference

Password protection enhancements

This new feature enhances the available password protection by storing passwords as hashed values. Storing passwords as one-way hashed values—rather than as encrypted values—more fully secures them because a malicious user can neither read nor decrypt them. You can select one of the following hashing algorithms:

- MD4: A one-way hash function that produces a 128-bit hash
- MD5: An improved, and more complex, version of MD4
- **SHA**: Secure Hash Algorithm, which produces a 160-bit hash, longer than MD5. The algorithm is slightly slower than MD5, but the larger message digest makes it more secure against brute-force collision and inversion attacks.

- **UNIX Crypt**: The UNIX encryption algorithm

- No Hashing

**See Also:**

- "Protection of User Passwords for Directory Authentication" on page 16-7 for a conceptual discussion
- Chapter 19, "Password Policies in Oracle Internet Directory" for instructions on setting password hashing

- **Replication tools**

The following new replication tools are now added:

- **Human Intervention Queue Manipulation tool**
  This tool enables you to move changes from the human intervention queue to either the retry queue or the purge queue.

- **OID Reconciliation Tool**
  This tool enables you to synchronize conflicting changes in a replicated environment.

**See Also:**

- "Using Command-Line Tools" on page 5-8 for a brief explanation of this tool
- "About the Human Intervention Queue Manipulation Tool" on page 30-38
- "About the Oracle Internet Directory Comparison and Reconciliation Tool" on page 30-38

- **Replication node deletion**

This new feature enables you to delete a node from a directory replication group.

**Tip:** "Deleting a Node from a Multimaster Replication Group" on page 30-18

- **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 Platform. See the chapter on concepts and components in the Oracle Identity Management Integration Guide
- **Upgrade procedures (release 2.1.1 only)**

  These new procedures enable you to upgrade from either Oracle Internet Directory release 2.0.4.x or release 2.0.6. Not supported in release 2.1.1.1 or in release 3.0.1.
Part I

Getting Started

Part I explains what Oracle Internet Directory is and some of the concepts you must know before using it. It contains these chapters:

- Chapter 1, "Links to Common Tasks"
- Chapter 2, "Introduction to LDAP and Oracle Internet Directory"
- Chapter 3, "Directory Concepts and Architecture"
- Chapter 4, "Post-Installation Tasks and Information"
- Chapter 5, "Directory Administration and Monitoring Tools"
- Chapter 6, "Process Control of Oracle Internet Directory Components"
Oracle Internet Directory routine administration tasks are described throughout this manual and in the tools chapters in Oracle Identity Management User Reference. This chapter points you to the information you need for some of the more common tasks. This chapter contains the following sections:

- Object Classes and Attributes
- Replication
- Security, Password Policies, and User Accounts
- Realms
- Server Processes, Instances, and Configuration Set Entries
- System Operational Attributes
- Naming Contexts
- Binds, Connections, Aliases, and Directory Discovery
- Referential Integrity
- Entries
- Groups
- Logging, Auditing, and Monitoring
- Tuning
- Garbage Collection
- Server Chaining and Data Migration
- Plug-ins

### Object Classes and Attributes

Oracle Internet Directory tasks related to object classes and attributes are shown in the following table.

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
</table>
| Add, modify, or delete object classes | "Managing Object Classes by Using Command-Line Tools" on page 11-7  
"Object Classes in the Directory" on page 11-2 |
Table 1–2  Replication

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set up replication</td>
<td>&quot;Preliminary Information for Installing and Configuring a Replication Group&quot; on page 30-2</td>
</tr>
<tr>
<td></td>
<td>&quot;Installing and Configuring Multimaster Replication&quot; on page 30-5</td>
</tr>
<tr>
<td></td>
<td>&quot;Installing and Configuring One-Way or Two-Way LDAP-Based Replication&quot; on page 30-19</td>
</tr>
<tr>
<td></td>
<td>Appendix I, &quot;Addition of a Directory Node by Using the Database Copy Procedure&quot;</td>
</tr>
<tr>
<td>Resolve replication conflicts</td>
<td>&quot;Resolving Conflicts Manually in a Replication Group&quot; on page 30-37</td>
</tr>
<tr>
<td>Configure replication failover</td>
<td>&quot;Configuring Replication Failover&quot; on page 30-43</td>
</tr>
<tr>
<td>Modify replication configuration parameters</td>
<td>&quot;Viewing and Modifying Directory Replication Server Configuration Parameters&quot; on page 31-1</td>
</tr>
<tr>
<td></td>
<td>&quot;Viewing and Modifying Parameters for Particular Replica Nodes&quot; on page 31-4</td>
</tr>
<tr>
<td></td>
<td>&quot;Modifying Parameters for Replication Agreements&quot; on page 31-6</td>
</tr>
<tr>
<td>Change the replication administrator’s password on all nodes</td>
<td>&quot;Changing the Replication Administrator’s Password on All Nodes Using Oracle Database Advanced Replication&quot; on page 31-10</td>
</tr>
<tr>
<td>Manage the change log</td>
<td>&quot;Managing the Change Log&quot; on page 31-10</td>
</tr>
<tr>
<td>Modify the speed of replication</td>
<td>&quot;Modifying the Speed of Directory Replication&quot; on page 31-11</td>
</tr>
<tr>
<td>Manage and monitor topology</td>
<td>&quot;Managing and Monitoring Topology&quot; on page 31-12</td>
</tr>
</tbody>
</table>

Security, Password Policies, and User Accounts

Oracle Internet Directory tasks related to security, password policies, and user accounts are shown in the following table.

Table 1–1  (Cont.) Object Classes and Attributes

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
</table>
Table 1–3  Security, Password Policies, and User Accounts

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set up SSL</td>
<td>Chapter 17, &quot;Secure Sockets Layer (SSL) and the Directory&quot;</td>
</tr>
<tr>
<td>View and modify access control information within ACPs</td>
<td>&quot;Managing Access Control by Using Oracle Directory Manager&quot; on page 18-14</td>
</tr>
<tr>
<td></td>
<td>&quot;Managing Access Control by Using Command-Line Tools&quot; on page 18-35</td>
</tr>
<tr>
<td>Manage password protection</td>
<td>&quot;Managing Password Protection by Using Oracle Directory Manager&quot; on page 20-3</td>
</tr>
<tr>
<td></td>
<td>&quot;Managing Password Protection by Using ldapmodify&quot; on page 20-3</td>
</tr>
<tr>
<td>Manage password verifiers for Oracle components</td>
<td>&quot;Managing Password Verifier Profiles for Oracle Components by Using Oracle Directory Manager&quot; on page 20-9</td>
</tr>
<tr>
<td></td>
<td>&quot;Managing Password Verifier Profiles for Oracle Components by Using Command-Line Tools&quot; on page 20-10</td>
</tr>
<tr>
<td>Use Oracle Directory Manager to view, refresh, and modify password policies</td>
<td>&quot;Managing Password Policies by Using Oracle Directory Manager&quot; on page 19-8</td>
</tr>
<tr>
<td>Manage password policies, accounts, and passwords by using command-line tools</td>
<td>&quot;Managing Password Policies, Accounts, and Passwords by Using Command-Line Tools&quot; on page 19-10</td>
</tr>
<tr>
<td>Manage Accounts and Passwords by Using the Self-Service Console</td>
<td>&quot;Managing Accounts and Passwords by Using the Self-Service Console&quot; on page 19-12</td>
</tr>
<tr>
<td>Set a user name or password for a super user, a guest user, or a proxy user</td>
<td>&quot;Managing Super Users, Guest Users, and Proxy Users by Using Oracle Directory Manager&quot; on page 7-9</td>
</tr>
<tr>
<td></td>
<td>&quot;Managing Super Users, Guest Users, and Proxy Users by Using ldapmodify&quot; on page 7-10</td>
</tr>
</tbody>
</table>

Realms

Oracle Internet Directory tasks related to realms are shown in the following table.

Table 1–4  Realms

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Customize the default realm.</td>
<td>&quot;Customizing the Default Identity Management Realm&quot; on page 23-12</td>
</tr>
<tr>
<td>Create additional realms</td>
<td>&quot;Creating Additional Identity Management Realms for Hosted Deployments&quot; on page 23-18</td>
</tr>
</tbody>
</table>

Server Processes, Instances, and Configuration Set Entries

Oracle Internet Directory tasks related to server processes, instances, and configuration set entries are shown in the following table.
<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Configure server instance parameters</td>
<td>&quot;Managing Server Configuration Set Entries by Using Command-Line Tools&quot; on 7-5</td>
</tr>
<tr>
<td></td>
<td>&quot;Managing Server Configuration Set Entries by Using Oracle Directory Manager&quot; on page 7-3</td>
</tr>
<tr>
<td>Start the directory server processes</td>
<td>&quot;Semantics of OPMN Starting Oracle Internet Directory&quot; on page 6-2</td>
</tr>
<tr>
<td>Stop the directory server processes</td>
<td>&quot;Semantics of OPMN Stopping Oracle Internet Directory&quot; on page 6-3</td>
</tr>
<tr>
<td>View system operational attributes</td>
<td>&quot;Setting System Operational Attributes by Using Oracle Directory Manager&quot; on page 7-7</td>
</tr>
<tr>
<td>Change the number of server processes or database connections by using Oracle Directory Manager</td>
<td>&quot;Changing the Configuration of the Default OID LDAP Server Instance&quot; on page 6-4</td>
</tr>
<tr>
<td>Add a configuration set to start additional Oracle Internet Directory LDAP server instances by using Oracle Directory Manager</td>
<td>&quot;Configuring Additional Oracle Internet Directory LDAP Server Instances&quot; on page 6-5</td>
</tr>
<tr>
<td>View server instances using Oracle Directory Manager.</td>
<td>&quot;Viewing Active Server Instance Information&quot; on page 7-11</td>
</tr>
<tr>
<td>Replace the default Oracle Internet Directory LDAP server instance with other instances by editing opmn.xml</td>
<td>&quot;Deconfiguring the Default Oracle Internet Directory LDAP Server Instance&quot; on page 6-5</td>
</tr>
<tr>
<td>Configure an instance of OID Replication Server by using the oidctl start command.</td>
<td>&quot;Configuring an Instance of the Oracle Internet Directory Replication Server&quot; on page 6-6</td>
</tr>
<tr>
<td>Configure an instance of the Oracle Internet Directory Server by using the oidctl start command.</td>
<td>&quot;Configuring an Oracle Directory Integration Platform Server Instance&quot; on page 6-6</td>
</tr>
<tr>
<td>View configuration set entries and their parameters by using Oracle Directory Manager.</td>
<td>&quot;Viewing Configuration Set Entries by Using Oracle Directory Manager&quot; on page 7-3</td>
</tr>
<tr>
<td>Add a configuration set entry.</td>
<td>&quot;Adding Configuration Set Entries by Using Oracle Directory Manager&quot; on page 7-4</td>
</tr>
<tr>
<td>Modify or delete an existing configuration set entry.</td>
<td>&quot;Adding Configuration Set Entries by Using Ldapadd&quot; on page 7-6 and the ldapmodify command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td></td>
<td>&quot;Modifying and Deleting Configuration Set Entries by Using Ldapmodify&quot; on page 7-6 and the ldapmodify command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td></td>
<td>&quot;Modifying Configuration Set Entries by Using Oracle Directory Manager&quot; on page 7-5</td>
</tr>
<tr>
<td></td>
<td>&quot;Deleting Configuration Set Entries by Using Oracle Directory Manager&quot; on page 7-5</td>
</tr>
</tbody>
</table>
Binds, Connections, Aliases, and Directory Discovery

System Operational Attributes

Oracle Internet Directory tasks related to system operational attributes are shown in the following table.

**Table 1–6  System Operational Attributes**

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>View or set operational attributes for an Oracle directory server to which you are connected by using Oracle Directory Manager.</td>
<td>&quot;Setting System Operational Attributes by Using Oracle Directory Manager&quot; on page 7-7</td>
</tr>
<tr>
<td>Modify system operational attributes by using <code>ldapmodify</code>.</td>
<td><code>ldapmodify</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
</tbody>
</table>

Naming Contexts

Oracle Internet Directory tasks related to naming contexts are shown in the following table.

**Table 1–7  Naming Contexts**

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable users to search for specific naming contexts.</td>
<td>&quot;Publishing Naming Contexts by Using Oracle Directory Manager&quot; on page 7-8</td>
</tr>
<tr>
<td>Back up and restore a small directory or naming context</td>
<td>&quot;Publishing Naming Contexts by Using <code>ldapmodify</code>&quot; on page 7-8</td>
</tr>
<tr>
<td></td>
<td>&quot;Backing Up and Restoring a Small Directory or Specific Naming Context&quot; on page 15-1</td>
</tr>
</tbody>
</table>

Binds, Connections, Aliases, and Directory Discovery

Oracle Internet Directory tasks related to binds, connections, aliases, and directory discovery are shown in the following table.

**Table 1–8  Binds, Connections, Aliases, and Directory Discovery**

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disable or enable anonymous binds</td>
<td>&quot;Managing Anonymous Binds&quot; on page 7-10</td>
</tr>
<tr>
<td>Authenticate a user or client to a directory server by using <code>ldapbind</code>.</td>
<td><code>ldapbind</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
<tr>
<td>Verify that you can connect a client to a server by using <code>ldapbind</code>.</td>
<td></td>
</tr>
<tr>
<td>Close idle LDAP connections.</td>
<td>&quot;Closing Idle LDAP Connections&quot; on page 7-11</td>
</tr>
<tr>
<td>Change the password to the Oracle Internet Directory database server.</td>
<td><code>oidpasswd</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
<tr>
<td>Add, modify, or search the directory with an alias entry by using command-line tools <code>ldapadd</code>.</td>
<td>&quot;About Alias Entries&quot; on page 7-12</td>
</tr>
</tbody>
</table>
Referential Integrity

Oracle Internet Directory tasks related to referential integrity are shown in the following table.

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Discover a directory</td>
<td>“Static Directory Server Discovery by Using the Directory Server</td>
</tr>
<tr>
<td></td>
<td>Usage File (ldap.ora)” on page 7-16</td>
</tr>
<tr>
<td></td>
<td>“Dynamic Directory Server Discovery by Using the Domain Name System (DNS)” on</td>
</tr>
<tr>
<td></td>
<td>page 7-17</td>
</tr>
<tr>
<td>Registering a directory</td>
<td>“Registering a Directory Server with the Domain Name System” on page 7-18</td>
</tr>
<tr>
<td>server with DNS.</td>
<td></td>
</tr>
</tbody>
</table>

Entries

Oracle Internet Directory tasks related to entries are shown in the following table.

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display or search for</td>
<td>“Searching for Entries by Using Oracle Directory Manager” on page 8-1</td>
</tr>
<tr>
<td>entries</td>
<td>The ldapsearch command-line tool reference in Oracle Identity</td>
</tr>
<tr>
<td></td>
<td>Management User Reference</td>
</tr>
<tr>
<td>View attributes for an</td>
<td>“Viewing Attributes for a Specific Entry by Using Oracle Directory Manager”</td>
</tr>
<tr>
<td>entry by using Oracle</td>
<td>on page 8-3</td>
</tr>
<tr>
<td>Directory Manager</td>
<td></td>
</tr>
<tr>
<td>Add an entry by using</td>
<td>“Adding Entries by Using Oracle Directory Manager” on page 8-3</td>
</tr>
<tr>
<td>Oracle Directory Manager</td>
<td></td>
</tr>
<tr>
<td>Modify an entry by using</td>
<td>“Modifying Entries by Using Oracle Directory Manager” on page 8-5</td>
</tr>
<tr>
<td>Oracle Directory Manager</td>
<td></td>
</tr>
<tr>
<td>Add, modify, or delete</td>
<td>“Managing Entries with Attribute Options by Using Oracle Directory Manager”</td>
</tr>
<tr>
<td>attribute options of an</td>
<td>on page 8-6</td>
</tr>
<tr>
<td>entry by using Oracle</td>
<td></td>
</tr>
<tr>
<td>Directory Manager</td>
<td></td>
</tr>
<tr>
<td>Manage content rules</td>
<td>“Managing Content Rules” on page 11-19</td>
</tr>
<tr>
<td>Add entries one at a time</td>
<td>The ldapadd command-line tool reference in Oracle Identity</td>
</tr>
<tr>
<td>by using ldapadd</td>
<td>Management User Reference</td>
</tr>
<tr>
<td>Configure a server with an</td>
<td></td>
</tr>
<tr>
<td>input file by using</td>
<td></td>
</tr>
<tr>
<td>ldapadd</td>
<td></td>
</tr>
<tr>
<td>Add several entries</td>
<td></td>
</tr>
<tr>
<td>concurrently by using</td>
<td></td>
</tr>
<tr>
<td>ldapaddmt</td>
<td></td>
</tr>
</tbody>
</table>
### Table 1–10 (Cont.) Entries

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compare attribute values you specify with those in a directory entry by using <code>ldapcompare</code>.</td>
<td>The <code>ldapcompare</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
<tr>
<td>Delete entries by using <code>ldapdelete</code>.</td>
<td>The <code>ldapdelete</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
<tr>
<td>Modify the DN or RDN of an entry by using <code>ldapmoddn</code>.</td>
<td>The <code>ldapmoddn</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
<tr>
<td>Rename an entry or a subtree by using <code>ldapmoddn</code>.</td>
<td>The <code>ldapmoddn</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
<tr>
<td>Move an entry or a subtree under a new parent by using <code>ldapmoddn</code>.</td>
<td>The <code>ldapmoddn</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
<tr>
<td>Create, update, and delete attribute data for an entry by using <code>ldapmodify</code>.</td>
<td>The <code>ldapmodify</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
<tr>
<td>Modify DN or RDN of an entry by using <code>ldapmodify</code>.</td>
<td>The <code>ldapmodify</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
<tr>
<td>Modify several entries concurrently by using <code>ldapmodifymt</code>.</td>
<td>The <code>ldapmodifymt</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
<tr>
<td>Import bulk data files</td>
<td>&quot;Importing an LDIF File by Using bulkload&quot; on page 9-4&lt;br&gt;The <code>bulkload</code> command-line tool reference in <em>Oracle Identity Management User Reference</em>&lt;br&gt;&quot;LDIF File Formatting Rules and Examples&quot; in <em>Oracle Identity Management User Reference</em></td>
</tr>
<tr>
<td>Convert directory data to LDIF by using <code>ldifwrite</code>.</td>
<td>&quot;ldifwrite Usage Examples&quot; on page 9-8&lt;br&gt;The <code>ldifwrite</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
<tr>
<td>Modify a large number of entries by using <code>bulkmodify</code>.</td>
<td>&quot;bulkmodify Usage Examples&quot; on page 9-7&lt;br&gt;The <code>bulkmodify</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
<tr>
<td>Delete a large number of entries by using <code>bulkdelete</code>.</td>
<td>&quot;bulkdelete Usage Examples&quot; on page 9-8&lt;br&gt;The <code>bulkdelete</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
<tr>
<td>Make an attribute searchable or non-searchable by using <code>catalog</code>.</td>
<td>&quot;catalog Usage Examples&quot; on page 9-10&lt;br&gt;The <code>catalog</code> command-line tool reference in <em>Oracle Identity Management User Reference</em></td>
</tr>
</tbody>
</table>

---

## Groups

Oracle Internet Directory tasks related to groups are shown in the following table.
Logging, Auditing, and Monitoring

Oracle Internet Directory tasks related to logging, auditing, and monitoring are shown in the following table.

Table 1–11 Groups

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manage static group entries by using Oracle Directory Manager</td>
<td>Managing Static Group Entries by Using Oracle Directory Manager on page 13-6</td>
</tr>
<tr>
<td>Manage static group entries by using command-line tools</td>
<td>&quot;Managing Static Group Entries by Using Command-Line Tools&quot; on page 13-7</td>
</tr>
<tr>
<td>Manage dynamic group entries by using Oracle Directory Manager</td>
<td>&quot;Managing Dynamic Groups by Using Oracle Directory Manager&quot; on page 13-9</td>
</tr>
<tr>
<td>Manage dynamic group entries by using command-line tools</td>
<td>&quot;Managing Dynamic Groups by Using Command-Line Tools&quot; on page 13-11</td>
</tr>
</tbody>
</table>

Table 1–12 Logging, Auditing, and Monitoring

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Limit logging to particular operations by specifying debug dimension</td>
<td>&quot;Setting the Operation Debug Dimension&quot; on page 14-6</td>
</tr>
<tr>
<td>Force flush debug messages to a log file</td>
<td>&quot;Force Flushing the Trace Information to a Log File&quot; on page 14-7</td>
</tr>
<tr>
<td>Set the audit level</td>
<td>&quot;Setting the Audit Level by Using Oracle Directory Manager&quot; on page 14-11</td>
</tr>
<tr>
<td>Search for audit log entries</td>
<td>&quot;Setting the Audit Level by Using ldapmodify&quot; on page 14-12</td>
</tr>
<tr>
<td>Configure the Oracle Internet Directory Server Manageability framework</td>
<td>&quot;Configuring Oracle Internet Directory Server Manageability&quot; on page 14-16</td>
</tr>
<tr>
<td>View Oracle Internet Directory server manageability information</td>
<td>&quot;Viewing Oracle Internet Directory Server Manageability Information&quot; on page 14-19</td>
</tr>
<tr>
<td>Set debug logging levels</td>
<td>&quot;Setting Debug Logging Levels&quot; on page 14-5</td>
</tr>
</tbody>
</table>

Tuning

Oracle Internet Directory tasks related to tuning are shown in the following table.

Table 1–13 Tuning

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimize Searches</td>
<td>&quot;Optimizing Searches for Large Group Entries&quot; on page 25-9</td>
</tr>
<tr>
<td></td>
<td>&quot;Optimizing Searches for Skewed Attributes&quot; on page 25-10</td>
</tr>
</tbody>
</table>
Garbage Collection

Oracle Internet Directory are shown in the following table.

### Table 1–14 Garbage Collection

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modify a garbage collector</td>
<td>&quot;Modifying a Garbage Collector by Using Oracle Directory Manager&quot; on page 26-6</td>
</tr>
<tr>
<td></td>
<td>&quot;Modifying a Garbage Collector by Using Command-Line Tools&quot; on page 26-7</td>
</tr>
<tr>
<td></td>
<td>&quot;Modifying the Oracle Internet Directory Statistics Collector&quot; on page 26-7</td>
</tr>
<tr>
<td>Enable, disable, or monitor logging of garbage collectors</td>
<td>&quot;Enabling Logging for Oracle Internet Directory Garbage Collectors&quot; on page 26-8</td>
</tr>
<tr>
<td></td>
<td>&quot;Disabling Logging for Oracle Internet Directory Garbage Collectors&quot; on page 26-8</td>
</tr>
<tr>
<td></td>
<td>&quot;Monitoring Garbage Collection Logging&quot; on page 26-8</td>
</tr>
</tbody>
</table>

Server Chaining and Data Migration

Oracle Internet Directory tasks related to server chaining and data migration are shown in the following table.

### Table 1–15 Server Chaining and Data Migration

<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Migrate data from an LDAP-compliant, third-party directory</td>
<td>&quot;Tasks For Migrating Data from LDAP-Compliant Directories&quot; on page 27-6</td>
</tr>
<tr>
<td>Migrate user data from an application-specific repository</td>
<td>&quot;Tasks For Migrating Data from Application-Specific Repositories&quot; on page 27-9</td>
</tr>
<tr>
<td>Configure Server Chaining</td>
<td>&quot;Configuring Server Chaining from the Command Line&quot; on page 28-3</td>
</tr>
<tr>
<td></td>
<td>&quot;Configuring Server Chaining by Using Oracle Directory Manager&quot; on page 28-4</td>
</tr>
</tbody>
</table>

Plug-ins

Oracle Internet Directory tasks related to plug-ins are shown in the following table.
<table>
<thead>
<tr>
<th>Task</th>
<th>Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create a plug-in</td>
<td>&quot;Creating Plug-ins&quot; on page 32-4</td>
</tr>
<tr>
<td>Register and manage plug-ins</td>
<td>&quot;Registering and Managing Plug-ins&quot; on page 32-4</td>
</tr>
<tr>
<td>Set up a password policy plug-in</td>
<td>Chapter 33, &quot;Oracle Internet Directory Plug-In for Password Policies&quot;</td>
</tr>
<tr>
<td>Set up an external authentication plug-in</td>
<td>Chapter 34, &quot;Setting Up the Customized External Authentication Plug-in&quot;</td>
</tr>
</tbody>
</table>
Introduction to LDAP and Oracle Internet Directory

This chapter introduces online directories, provides an overview of the Lightweight Directory Application Protocol (LDAP) version 3, and explains some of the unique features and benefits of Oracle Internet Directory.

This chapter contains these topics:

- What Is a Directory?
- What Is the Lightweight Directory Access Protocol (LDAP)?
- Oracle Identity Management
- What Is Oracle Internet Directory?
- How Oracle Components Use Oracle Internet Directory

What Is a Directory?

A directory is a way in which complex information is organized, making it easy to find. Directories list resources—for example, people, books in a library, or merchandise in a department store—and give details about each one. They can be either offline—for example, a telephone book or a department store catalog—or online.

Online directories are used by enterprises with distributed computer systems for fast searches, cost-effective management of users and security, and as central integration points for multiple applications and services. Online directories are also becoming critical to both e-businesses and hosted environments.

This section contains these topics:

- The Expanding Role of Online Directories
- The Problem: Too Many Special-Purpose Directories

The Expanding Role of Online Directories

An online directory is a specialized database that stores and retrieves collections of information about objects. Such information can represent any resources that require management: employee names, titles, and security credentials; information about partners; or information about shared network resources such as conference rooms and printers.

Online directories can be used by a variety of users and applications, and for a variety of purposes, including:
What Is a Directory?

- An employee searching for corporate white page information, and, through a mail client, looking up e-mail addresses
- An application, such as a message transport agent, locating a user’s mail server
- A database application identifying role information for a user

Although an online directory is a database—that is, a structured collection of data—it is not a relational database. The following table contrasts online directories with relational databases.

### Table 2–1 Comparison of Online Directories and Relational Databases

<table>
<thead>
<tr>
<th>Online Directories</th>
<th>Relational Databases</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Primarily read-focused.</strong> Typical use involves a relatively small number of data updates, and a potentially large number of data retrievals.</td>
<td><strong>Primarily write-focused.</strong> Typical use involves continuous recording of transactions, with retrievals done relatively infrequently.</td>
</tr>
<tr>
<td>Designed to handle relatively simple transactions on relatively small units of data. For example, an application might use a directory simply to store and retrieve an e-mail address, a telephone number, or a digital portrait.</td>
<td>Designed to handle large and diverse transactions using many operations on large units of data.</td>
</tr>
<tr>
<td>Designed to be location-independent. Directory-enabled applications expect, at all times, to see the same information throughout the deployment environment—regardless of which server they are querying. If a queried server does not store the information locally, then it must either retrieve the information or point the client application to it transparently.</td>
<td>Typically designed to be location-specific. While a relational database can be distributed, it usually resides on a particular database server.</td>
</tr>
<tr>
<td>Designed to store information in entries. These entries might represent any resource customers wish to manage: employees, e-commerce partners, conference rooms, or shared network resources such as printers. Associated with each entry is a number of attributes, each of which may have one or more values assigned. For example, typical attributes for a person entry might include first and last names, e-mail addresses, the address of a preferred mail server, passwords or other login credentials, or a digitized portrait.</td>
<td>Designed to store information as rows in relational tables.</td>
</tr>
</tbody>
</table>

**The Problem: Too Many Special-Purpose Directories**

According to some estimates, each of the world’s largest companies has an average of 180 different directories, each designated for a special purpose. Add to this the various enterprise applications, each with its own additional directory of user names, and the actual number of special purpose directories becomes even greater.

Managing so many special purpose directories can cause problems:

- High cost of administration: Administrators must maintain essentially the same information in many different places. For example, when an enterprise hires a new employee, administrators must create a new user identity on the network, create a new e-mail account, add the user to the human-resources database, and set up all applications that the employee may need—for example, user accounts on development, testing, and production database systems. Later, if the employee...
leaves the company, administrators must reverse the process to disable all these user accounts.

- Inconsistent data: Because of the large administrative overhead, it can be difficult for multiple administrators, entering redundant information in multiple systems, to synchronize this employee information across all systems. The result can be inconsistent data across the enterprise.

- Security issues: Each separate directory may have its own password policy—which means that a user may struggle with a variety of user names and passwords, each for a different system.

Today’s enterprises need a more general purpose directory infrastructure, one based on a common standard for supporting a wide variety of applications and services.

What Is the Lightweight Directory Access Protocol (LDAP)?

LDAP is a standard, extensible directory access protocol. It is a common language that LDAP clients and servers use to communicate.

This section contains these topics:

- LDAP and Simplified Directory Management
- LDAP Version 3

LDAP and Simplified Directory Management

LDAP was conceived as an Internet-ready, lightweight implementation of the International Standardization Organization (ISO) X.500 standard for directory services. It requires a minimal amount of networking software on the client side, which makes it particularly attractive for Internet-based, thin client applications.

The LDAP standard simplifies management of directory information in three ways:

- It provides all users and applications in the enterprise with a single, well-defined, standard interface to a single, extensible directory service. This makes it easier to rapidly develop and deploy directory-enabled applications.

- It reduces the need to enter and coordinate redundant information in multiple services scattered across the enterprise.

- Its well-defined protocol and array of programmatic interfaces make it more practical to deploy Internet-ready applications that leverage the directory.

LDAP Version 3

The most recent version of LDAP, Version 3, was approved as a proposed Internet Standard by the Internet Engineering Task Force (IETF) in December 1997. LDAP Version 3 improves on LDAP Version 2 in several important areas:

- Globalization Support: LDAP Version 3 allows servers and clients to support characters used in every language in the world.

- Knowledge references (also called referrals): LDAP Version 3 implements a referral mechanism that allows servers to return references to other servers as a result of a directory query. This makes it possible to distribute directories globally by partitioning a directory information tree (DIT) across multiple LDAP servers.
■ Security: LDAP Version 3 adds a standard mechanism for supporting Simple Authentication and Security Layer (SASL), providing a comprehensive and extensible framework for data security.

■ Extensibility: LDAP Version 3 enables vendors to extend existing LDAP operations through the use of mechanisms called controls. These are extra pieces of information carried along with existing operations, altering the behavior of the operation. When a client application passes a control along with the standard LDAP command, the behavior of the commanded operation is altered accordingly. For example, when a client wants to modify meta-information hidden in the directory, it can send the manageDSAIT control along with the LDAP command.

■ Feature and schema discovery: LDAP Version 3 enables publishing information useful to other LDAP servers and clients, such as the supported LDAP protocols and a description of the directory schema.

See Also:
- RFCs (Requests for Comments) 2251-2256 of the IETF, available at: http://www.ietf.org
- Appendix K, "RFCs Supported by Oracle Internet Directory"
- "Related Documents" on page xxxvi for an additional list of resources on LDAP
- Chapter 3, "Directory Concepts and Architecture" for a conceptual discussion of directory information trees and knowledge references
- "About LDAP Controls" in Oracle Identity Management User Reference for a list and description of controls supported by Oracle Internet Directory

Oracle Identity Management

Oracle Internet Directory is a component of Oracle Identity Management, an integrated infrastructure that provides distributed security services for Oracle products and other enterprise applications. In addition to Oracle Internet Directory, the Oracle Identity Management infrastructure includes the following components and capabilities:

■ Oracle Directory Integration Platform: This component enables synchronization between Oracle Internet Directory and:
  - Other directories and user repositories
  - Automatic provisioning services for Oracle components and applications
  - Third-party applications

■ Oracle Delegated Administration Services: This component provides trusted proxy-based administration of directory information by users and application administrators.

■ Oracle Application Server Single Sign-On: This component provides single sign-on access to Oracle and third-party Web applications.

■ Oracle Application Server Certificate Authority: This component generates and publishes X.509 V3 PKI certificates to support strong authentication methods.

To support enterprise application deployments, a single Oracle Identity Management infrastructure is typically deployed in the enterprise. It can include multiple server
and component instances to provide high availability, information localization, and delegated component administration. Each additional application in the enterprise then leverages the shared infrastructure for identity management services. This deployment model has a number of advantages, including:

- Planning and implementing the identity management infrastructure is a one-time cost, rather than a necessary part of each enterprise application deployment. As a result, new applications such as portals, J2EE applications, and e-business applications can be rapidly deployed.
- Identities, while possibly administered in multiple places, are centrally managed and instantly available to all enterprise applications.
- A centralized security infrastructure makes it possible to realize user single sign-on across enterprise applications.
- A centralized identity management infrastructure provides a single point of integration between the enterprise Oracle environment and other identity management systems. This eliminates the need for multiple, custom, point-to-point integration solutions.

See Also:
- *Oracle Identity Management Concepts and Deployment Planning Guide* for information about planning, deploying and using the Oracle Identity Management infrastructure

What Is Oracle Internet Directory?

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

This section contains these topics:

- Overview of Oracle Internet Directory
- Components of Oracle Internet Directory
- Advantages of Oracle Internet Directory

Overview of Oracle Internet Directory

Oracle Internet Directory runs as an application on an Oracle Database. It communicates with the database by using Oracle Net Services, Oracle's operating system-independent database connectivity solution. The database may or may not be on the same host. Figure 2–1 illustrates this relationship.
What Is Oracle Internet Directory?

Oracle Internet Directory includes:

- Oracle directory server, which responds to client requests for information about people and resources, and to updates of that information, by using a multitiered architecture directly over TCP/IP

- Oracle directory replication server, which replicates LDAP data between Oracle directory servers

- Directory administration tools, which include:
  - Oracle Directory Manager, which simplifies directory administration through a Java-based graphical user interface
  - A variety of command-line administration and data management tools invoked from LDAP clients
  - Directory server management tools within Oracle Enterprise Manager 10g Application Server Control Console. These tools enable you to:
    * Monitor real-time events and statistics from a normal browser
    * Start the process of collecting such data into a new repository

- Oracle Internet Directory Software Developer’s Kit

See Also: Oracle Identity Management Application Developer’s Guide for information about the Oracle Internet Directory Software Developer's Kit
Advantages of Oracle Internet Directory

Among its more significant benefits, Oracle Internet Directory provides scalability, high availability, security, and tight integration with the Oracle environment.

Scalability

Oracle Internet Directory exploits the strengths of an Oracle Database, enabling support for terabytes of directory information. In addition, such technologies as shared LDAP servers and database connection pooling enable it to support thousands of concurrent clients with subsecond search response times.

Oracle Internet Directory also provides data management tools, such as Oracle Directory Manager and a variety of command-line tools, for manipulating large volumes of LDAP data.

High Availability

Oracle Internet Directory is designed to meet the needs of a variety of important applications. For example, it supports full multimaster replication between directory servers: If one server in a replication community becomes unavailable, then a user can access the data from another server. Information about changes to directory data on a server is stored in special tables on the Oracle Database. These are replicated throughout the directory environment by Oracle Database Advanced Replication, a robust replication mechanism.

Oracle Internet Directory also takes advantage of all the availability features of the Oracle Database. Because directory information is stored securely in the Oracle Database, it is protected by Oracle’s backup capabilities. Additionally, the Oracle Database, running with large data stores and heavy loads, can recover from system failures quickly.

Security

Oracle Internet Directory offers comprehensive and flexible access control. An administrator can grant or restrict access to a specific directory object or to an entire directory subtree. Moreover, Oracle Internet Directory implements three levels of user authentication: anonymous, password-based, and certificate-based using Secure Sockets Layer (SSL) Version 3 for authenticated access and data privacy.

Integration with the Oracle Environment

Through Oracle Directory Integration Platform, Oracle Internet Directory provides a single point of integration between the Oracle environment and other directories such as NOS directories, third-party enterprise directories, and application-specific user repositories.

How Oracle Components Use Oracle Internet Directory

Oracle components use Oracle Internet Directory for easier administration, tighter security, and simpler integration between multiple directories.

This section contains these topics:

- Easier and More Cost-Effective Administration of Applications
- Tighter Security Through Centralized Security Policy Administration
- Integration of Multiple Directories
Easier and More Cost-Effective Administration of Applications

**OracleAS Portal** enables self-service, integrated enterprise portals to store common user and group attributes in Oracle Internet Directory. The Oracle Portal administration tool also leverages the Oracle Delegated Administration Services for certain tasks.

**Oracle Collaboration Suite** uses Oracle Internet Directory for:

- Centralized management of information about users and groups
- Provisioning Oracle Collaboration Suite components—that is, notifying them whenever changes of interest are applied to data in Oracle Internet Directory
- Centralized integration for enterprises connecting other directories with any Oracle Collaboration Suite component

**Oracle Net Services** uses Oracle Internet Directory to store and resolve database services and the simple names, called net service names, that can be used to represent them.

Tighter Security Through Centralized Security Policy Administration

The **Oracle Database** uses Oracle Internet Directory to store user names and passwords. It uses Oracle Internet Directory to store a password verifier along with the entry of each user.

Oracle Application Server Single Sign-On uses Oracle Internet Directory to store user entries. It maps users for any partner application to entries in Oracle Internet Directory, and authenticates those users by using LDAP mechanisms.

Oracle Advanced Security uses Oracle Internet Directory for:

- Central Management of user authentication credentials
  
  Instead of storing a user's database password in each database, Oracle Advanced Security stores it in one place: the directory. It stores the password as an attribute of the user entry.

- Central management of user authorizations
  
  Oracle Advanced Security uses directory entries, called enterprise roles, to determine the privileges for a given enterprise user within a given schema, whether that schema is shared or owned. Enterprise roles are containers for database-specific global roles. For example, a user might be assigned the enterprise role of clerk, which might contain the global role of hrclerk with its attendant privileges on the human resources database and the global role of analyst with its attendant privileges on the payroll database.

- Mappings to shared schemas
  
  Oracle Advanced Security uses mappings—that is, directory entries that point an enterprise user to shared application schemas on the database instead of to an individual account. For example, you might map several enterprise users to the schema `sales_application` instead of to separate accounts in their names.

- Single password authentication
  
  In the Oracle Database, Oracle Advanced Security enables enterprise users to authenticate to multiple databases by using a single, centrally managed password. The password is stored in the directory as an attribute of the user's entry and is protected by encryption and access control lists. This spares you from setting up
Secure Sockets Layer (SSL) on clients and users from having to remember multiple passwords.

- Enterprise user security

The alternative to authenticating with a centrally managed password is to use PKI-based enterprise user security through SSL. Like single password authentication, this feature relies on a user entry in the directory. A user’s wallet must be stored as an attribute of his or her entry.

- Central storage of PKI credentials

In Oracle Database and Oracle Application Server, user wallets can be stored in the directory as an attribute of the user’s entry. This enables mobile users to retrieve and open their wallets by using Enterprise Login Assistant. While the wallet is open, authentication is transparent—that is, users can access any database on which they own or share a schema without having to authenticate again.

**Integration of Multiple Directories**

Oracle Directory Integration Platform is a collection of interfaces and services for integrating multiple directories by using Oracle Internet Directory and several associated plug-ins and connectors. It provides these benefits:

- All Oracle components are pre-certified to work with Oracle Internet Directory.

- You can integrate the entire Oracle environment with third-party directories simply by integrating each third-party directory with Oracle Internet Directory. This saves you from having to integrate each application with each directory.
This chapter provides conceptual descriptions of the basic elements of Oracle Internet Directory and discusses Oracle Internet Directory architecture.

This chapter contains these topics:

- Oracle Internet Directory Architecture
- Example: How Oracle Internet Directory Works
- Entries
- Attributes
- Object Classes
- Naming Contexts
- Security
- Globalization Support
- Distributed Directories
- Knowledge References and Referrals
- Oracle Delegated Administration Services and the Oracle Internet Directory Self-Service Console
- The Service Registry and Service to Service Authentication
- Oracle Directory Integration Platform
- Oracle Internet Directory and Identity Management
- Resource Information

**See Also:** "Related Documents" on page xxxvi for suggestions on further reading about LDAP-compliant directories

### Oracle Internet Directory Architecture

This section contains these topics:

- An Oracle Internet Directory Node
- An Oracle Directory Server Instance
- Directory Metadata
- Configuration Set Entries
An Oracle Internet Directory Node

An Oracle Internet Directory node consists of one or more directory server instances connected to the same directory store. The directory store—that is, the repository of the directory data—is an Oracle Database.

Figure 3–1 on page 3-2 shows the various directory server components and their relationships running on a single node.

Oracle Net Services is used for all connections between the Oracle database server and:

- The object class
- The Oracle directory server instance 1 non-SSL port 389
- The Oracle directory server instance 2 SSL-enabled port 636
- The OID Monitor

LDAP is used for connections between directory server instance 1 on non-SSL port 389 and:

- Oracle Directory Manager
- Oracle directory replication server

The two Oracle directory server instances and the Oracle directory replication server connect to OID Monitor by way of the operating system.

As shown in Figure 3–1, an Oracle Internet Directory node includes the following major components:
The Oracle directory replication server uses LDAP to communicate with an Oracle directory (LDAP) server instance. To communicate with the database, all components use OCI/Oracle Net Services. Oracle Directory Manager and the command-line tools communicate with the Oracle directory servers over LDAP.

### An Oracle Directory Server Instance

Each Oracle directory server instance, also called an LDAP server instance, looks similar to what Figure 3–2 illustrates.
One instance comprises one dispatcher process and one or more server processes. By default, there is one server process for each instance, but you can increase this number. Oracle Internet Directory dispatcher and server processes can use multiple threads to distribute the load. LDAP clients send LDAP requests to an Oracle Internet Directory listener/dispatcher process listening for LDAP commands at its port.

The Oracle Internet Directory listener/dispatcher sends the request to the Oracle directory server which, in turn, creates server processes. A server process handles an LDAP operation request and connects to the Oracle database instance to access the directory store. The directory server handles the client request by generating one server process for each operation.

Multiple server processes enable Oracle Internet Directory to take advantage of multiple processor systems. The number of server processes created is determined by the configuration parameter ORCLSERVERPROCS. The default is 1 (one).

Database connections from each server process are spawned as needed, depending on the value set for the configuration parameter ORCLMAXCC. The number of database connections spawned by each server is equal to ORCLMAXCC + (ORCLMAXCC/2) + 1. The default value of ORCLMAXCC in configset0 is 2. The server processes communicate with the data server by way of Oracle Net Services. an Oracle Net Services Listener/Dispatcher relays the request to the Oracle Database.

**Directory Metadata**

Directory metadata is the information used by the directory server during run time for processing LDAP requests. It is stored in the underlying data repository. During startup, the directory server reads this information and stores it in a local metadata cache. It then uses this cache during its runtime to process incoming LDAP operation requests.

The directory server has the following types of metadata in its local metadata cache:

- **Directory Schema**

  The definitions of object classes, attributes, and matching rules supported by the directory server. The directory server uses this information during creation and
modification of directory objects. A directory object is a collection of object classes and their associated attributes and matching rules.

- **Access control policy point (ACP)**
  A directory administrative domain for defining and controlling access to the information in that domain. The directory server uses ACPs when determining whether to allow a certain LDAP operation performed by a user.

- **Root DSE entry**
  The root DSE (DSA-Specific Entry) contains a number of attributes that store information about the directory server itself. For example, these attributes contain the following information items, to mention just a few:
  - Naming contexts DNs
  - Sub Schema Subentry DN
  - Superior references (referrals) DNs
  - Special entry DNs like Oracle Internet Directory configuration and registry containers
  - Special Entry DNs like change log and change status containers
  - DN of replications agreement container

- **Privilege groups**
  Groups that can be used in access control policies.
  The directory schema supports directory group objects through the standard groupofuniquenames and groupofnames object classes. These object classes hold information for such groups as distribution lists and mailing lists to mention just two.

  Oracle Internet Directory extends these standard group objects through an auxiliary object class called orclprivilegegroup. This object class, which supports privilege groups that can be used in access control policies, provides flexibility to grant or deny access to groups of users. The directory server uses this information during:
  - LDAP bind operations to find out the subscribed privileged groups for a given user
  - Access control policy evaluation if the policy has directives that grant or deny access to privileged groups

  **See Also:**
  - "Managing Entries by Using Oracle Directory Manager” on page 8-1
  - "Example: Adding a User Entry by Using Oracle Directory Manager” on page 8-4

- **Catalog entry**
  A special entry containing information about indexed attributes in the underlying database. The directory uses this information during directory search operations.

- **Common entry**
  A special entry containing information about hosted companies. A hosted company is an enterprise to which another enterprise provides services. The
metadata in this entry includes the hosted company DN, user search base, nickname and other attributes, all of which are described in Chapter 23, "Deployment of Oracle Identity Management Realms".

■ Plug-in entry
   A special entry containing information about the kind of operation that triggers a plug-in event, and the point in the operation when that plug-in is to be triggered. This information is described in Chapter 32, "Oracle Internet Directory Server Plug-in Framework".

■ Password verifier entry
   A special entry containing information about the encryption and verifier attribute types. This information is described in Chapter 20, "Directory Storage of Password Verifiers".

■ Password policy entry
   A special entry containing information about the policies enforced by the directory server for the user password credentials. The directory server uses this information during runtime to enforce the password policies.

**Configuration Set Entries**

The configuration parameters for each Oracle directory server instance are stored in an entry called a configuration set entry, or configset. When you start an instance of a server by using the OID Control Utility, the start-command you enter contains a reference to one of these configuration set entries and uses the information it contains.

The Oracle directory server is installed with a default configuration set entry (configset0) so that you can run the directory server immediately. You can create customized configuration set entries with parameters to meet your specific needs.

You can view, add, and modify configuration set entries by using either Oracle Directory Manager or the appropriate command-line tool.

**See Also:**

- "Managing Server Configuration Set Entries" on page 7-1
- "Oracle Internet Directory Configuration Schema Elements" in Oracle Identity Management User Reference for a list of configuration set entry attributes

**Example: How Oracle Internet Directory Works**

This example shows you how Oracle Internet Directory processes a search request.

1. The user or client enters a search request that is conditioned by one or more of the following options:
   - SSL: The client and server can establish a session that uses SSL encryption and authentication, or SSL encryption only. If SSL is not used, the client's message is sent in clear text.
   - Type of user: The user can seek access to the directory either as a particular user or as an anonymous user, depending on which of the two has the necessary privileges to perform the desired function.
Filters: The user can narrow the search by using one or more search filters, including those that use the Boolean conditions "and," "or," and "not," and those that use other operators such as "greater than," "equal to," and "less than."

2. If the user or client issues the command by using Oracle Directory Manager, then the latter invokes a query function in the Java Native Interface which, in turn, invokes a function in the C API. If the user or client uses a command-line tool, then the tool directly invokes a C function in the C API.

3. The C API, using the LDAP protocol, sends a request to a directory server instance to connect to the directory.

4. The directory server authenticates the user, a process called binding. The directory server also checks the Access Control Lists (ACLs) to verify that the user is authorized to perform the requested search.

5. The directory server converts the search request from LDAP to Oracle Call Interface (OCI)/Oracle Net Services and sends it to the Oracle Database.

6. The Oracle Database retrieves the information and passes it back through the chain—to the directory server, then to the C API, and, finally, to the client.

Entries

In an online directory, each collection of information about an object is called an entry. An entry can include, for example, information about an employee, a conference room, an e-commerce partner, or a shared network resource such as a printer.

This section contains these topics:

- Distinguished Names (DNs) and Directory Information Trees (DITs)
- Entry Caching

Distinguished Names (DNs) and Directory Information Trees (DITs)

Each entry in an online directory is uniquely identified by a distinguished name (DN). The distinguished name tells you exactly where the entry resides in the directory hierarchy. This hierarchy is represented by a directory information tree (DIT).

To understand the relation between a distinguished name and a directory information tree, look at Figure 3–3.

Figure 3–3  A Directory Information Tree

The DIT in Figure 3–3 includes entries for two employees of Acme Corporation who are both named Anne Smith. It is structured along geographical and organizational
lines. The Anne Smith contained in the left branch works in the Sales division in the United States, while the other works in the Server Development division in the United Kingdom.

The Anne Smith contained in the right branch has the common name (cn) Anne Smith. She works in an organizational unit (ou) named Server Development, in the country (c) of United Kingdom of Great Britain and Northern Ireland (uk), in the organization (o) Acme.

The DN for this "Anne Smith" entry is:

cn=Anne Smith,ou=Server Development,c=uk,o=acme

Note that the conventional format of a distinguished name places the lowest DIT component at the left, then follows it with the next highest component, moving progressively up to the root.

Within a distinguished name, the lowest component is called the relative distinguished name (RDN). For example, in the previous entry for Anne Smith, the RDN is cn=Anne Smith. Similarly, the RDN for the entry immediately above Anne Smith's RDN is ou=Server Development, the RDN for the entry immediately above ou=Server Development is c=uk, and so on. A DN is thus a concatenation of RDNs that reflects parent-child relationships in the DIT. Within the DN, RDNs are separated by commas.

To locate a particular entry within the overall DIT, a client uniquely identifies that entry by using the full DN—not simply the RDN—of that entry. For example, within the global organization in Figure 3–3, to avoid confusion between the two Anne Smiths, you would use each one's full DN. If there are potentially two employees with the same name in the same organizational unit, you could use additional mechanisms—for example, you could identify each employee with a unique number.

**Entry Caching**

To make operations on entries quick and efficient, Oracle Internet Directory uses entry caching. When you enable this feature, Oracle Internet Directory assigns a unique identifier to each entry, then stores a specified number of those identifiers in cache memory. When a user performs an operation on an entry, the directory server looks in the cache for the entry identifier, then retrieves the corresponding entry from the directory. This method enhances Oracle Internet Directory performance, and is especially useful in smaller and medium-sized enterprises.

---

**Note:** In Oracle Internet Directory 10g (10.1.4.0.1), you can use entry caching only in the case of a single server, single instance Oracle Internet Directory node.

**See Also:** Chapter 8, "Directory Entries Administration"

**Attributes**

In a typical telephone directory, an entry for a person contains such information items as an address and a phone number. In an online directory, such an information item is called an attribute. Attributes in a typical employee entry can include, for example, a job title, an e-mail address, or a phone number.

For example, in Figure 3–4, the entry for Anne Smith in Great Britain (uk) has several attributes, each providing specific information about her. These are listed in the
balloon to the right of the tree, and they include emailaddrs, printername, jpegPhoto, and app preferences. Moreover, each bullet in Figure 3–4 is also an entry with attributes, although the attributes for each are not shown.

*Figure 3–4  Attributes of the Entry for Anne Smith*

Each attribute consists of an attribute type and one or more attribute values. The attribute type is the kind of information that the attribute contains—for example, jobTitle. The attribute value is the particular occurrence of information appearing in that entry. For example, the value for the jobTitle attribute could be manager.

This section contains these topics:

- Kinds of Attribute Information
- Single-Valued and Multivalued Attributes
- Common LDAP Attributes
- Attribute Syntax
- Attribute Matching Rules
- Attribute Options

**Kinds of Attribute Information**
Attributes contain two kinds of information.

- Application Attributes
  This information is maintained and retrieved by directory clients and is unimportant to the operation of the directory. A telephone number, for example, is application information.

- Operational Attributes
  This information 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 the creation or modification of an entry, or the name of the user who creates or modifies an entry. Other operational information, such as access information, is defined by administrators and is used by the directory program in its processing.

To enhance your ability to search for entries, Oracle Internet Directory automatically creates several system operational attributes when you add an entry to the directory. These include:
Moreover, when a user modifies an entry, Oracle Internet Directory automatically updates the `modifiersName` and `modifyTimestamp` attributes to, respectively, the name of the person modifying the entry, and the time of the entry modification in UTC.

See Also: "Setting System Operational Attributes" on page 7-7 for instructions on configuring system operational attributes

### Single-Valued and Multivalued Attributes

Attributes can be either single-valued or multivalued. Single-valued attributes carry only one value in the attribute, whereas multivalued attributes can have several. An example of a multivalued attribute is a group membership list with names of everyone in the group.

### Common LDAP Attributes

Oracle Internet Directory implements all of the standard LDAP attributes. Some of the more common ones defined by RFC 2798 of the Internet Engineering Task Force (IETF) are shown in Table 3–3.

#### Table 3–3 Common LDAP Attributes

<table>
<thead>
<tr>
<th>Attribute Type</th>
<th>Attribute String</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>commonName</td>
<td>cn</td>
<td>Common name of an entry—for example, Anne Smith.</td>
</tr>
<tr>
<td>domainComponent</td>
<td>dc</td>
<td>The DN of the component in a Domain Name System (DNS)—for example, dc=uk, dc=acme, dc=com.</td>
</tr>
<tr>
<td>jpegPhoto</td>
<td>jpegPhoto</td>
<td>Photographic image in JPEG format. This is stored in binary format.</td>
</tr>
<tr>
<td>organization</td>
<td>o</td>
<td>Name of an organization—for example, my_company.</td>
</tr>
<tr>
<td>organizationalUnitName</td>
<td>ou</td>
<td>Name of a unit within an organization—for example, Server Development.</td>
</tr>
<tr>
<td>owner</td>
<td>owner</td>
<td>Distinguished name of the person who owns the entry, for example, cn=Anne Smith, ou=Server Development, o= Acme, c=uk.</td>
</tr>
<tr>
<td>surname, sn</td>
<td>sn</td>
<td>Last name of a person—for example, Smith.</td>
</tr>
<tr>
<td>telephoneNumber</td>
<td>telephoneNumber</td>
<td>Telephone number—for example, (650) 123-4567 or 6501234567.</td>
</tr>
</tbody>
</table>

See Also: "Oracle Identity Management LDAP Attribute Reference" in Oracle Identity Management User Reference for a list of several attributes Oracle Internet Directory provides.
Attribute Syntax

Attribute syntax is the format of the data that can be loaded into each attribute. For example, the syntax of the `telephoneNumber` attribute might require a telephone number to be a string of numbers containing spaces and hyphens. However, the syntax for another attribute might require specifying whether the data has to be in the form of a date, or whether the data can consist of numbers only. Each attribute must have one and only one syntax.

Oracle Internet Directory recognizes most of the syntaxes specified in RFC 2252 of the Internet Engineering Task Force (IETF), allowing you to associate most of the syntaxes described in that document with an attribute. In addition to recognizing the syntaxes in RFC 2252, Oracle Internet Directory also enforces some LDAP syntaxes. You cannot add new syntaxes beyond those already supported by Oracle Internet Directory.

**See Also:** "About LDAP Attribute Syntax" in Oracle Identity Management User Reference

Attribute Matching Rules

In response to most incoming client requests, the directory server performs search and compare operations. During these operations, the directory server consults the relevant matching rule to determine 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.

Oracle Internet Directory implements all the standard LDAP matching rules. You cannot add new matching rules beyond those already supported by Oracle Internet Directory.

**See Also:** "About LDAP Matching Rules" in Oracle Identity Management User Reference

Attribute Options

An attribute type can have various options that enable you to specify how the value for that attribute is made available in a search or a compare operation. For example, suppose that an employee has two addresses, one in London, the other in New York. Options for that employee’s `address` attribute could allow you to store both addresses.

Moreover, attribute options can include language codes. For example, options for John Doe’s `givenName` attribute could enable you to store his given name in both French and Japanese.

For clarity, we can distinguish between an attribute with an option and its base attribute, which is the same attribute without an option. For example, in the case of `givenName; lang-fr=Jean`, the base attribute is `givenName`; the French value for that base attribute is `givenName; lang-fr=Jean`.

An attribute with one or more options inherits the properties—for example, matching rules and syntax—of its base attribute. To continue the previous example, the attribute with the option `cn; lang-fr=Jean` inherits the properties of `cn`.

**See Also:** "About LDAP Attribute Syntax" in Oracle Identity Management User Reference
Object Classes

An object class is a group of attributes that define the structure of an entry. When you define a directory entry, you assign one or more object classes to it. Some of the attributes in these object classes are mandatory and must have values, others are optional and can be empty.

For example, the organizationalPerson object class includes the mandatory attributes commonName (cn) and surname (sn), and the optional attributes telephoneNumber, uid, streetAddress, and userPassword. When you define an entry by using the organizationalPerson object class, you must specify values for commonName (cn) and surname (sn). You do not need to provide values for telephoneNumber, uid, streetAddress, and userPassword.

This section contains these topics:

- Subclasses, Superclasses, and Inheritance
- Object Class Types

Subclasses, Superclasses, and Inheritance

A subclass is an object class derived from another object class. The object class from which a subclass is derived is called its superclass. For example, the object class organizationalPerson is a subclass of the object class person. Conversely, the object class person is the superclass of the object class organizationalPerson.

Subclasses inherit all of the attributes belonging to their superclasses. For example, the subclass organizationalPerson inherits the attributes of its superclass, person. Entries also inherit attributes that their superclasses have inherited.

Note: In itself, an object class contains no values. Only an instance of an object class— that is, an entry—contains values. When a subclass inherits attributes from a superclass, it inherits only the attribute definitions of the superclass.

One special object class, called top, has no superclasses. It is one of the superclasses of every object class in the directory, and its attribute definitions are inherited by every entry.

Object Class Types

There are three types of object classes:

- Subclasses, Superclasses, and Inheritance
- Object Class Types
Structural Object Classes

Structural object classes describe the basic aspects of an object. Most of the object classes that you use are structural object classes, and every entry should belong to at least one structural object class. Examples of structural object classes are person and groupOfNames.

These object classes model real-world entities and their physical or logical attributes. Examples include people, printers, and database connections.

Structural object classes use structure rules to place restrictions on the kinds of objects you can create under any given object class. For example, a structure rule might require all objects below the organization (o) object class to be organizational units (ou). Following this rule, you could not enter person objects directly below an organization object class. Similarly, a structure rule might disallow you from placing an organizational unit (ou) object below a person object.

Auxiliary Object Classes

Auxiliary object classes are groupings of optional attributes that expand the existing list of attributes in an entry. Unlike structural object classes, they do not place restrictions on where an entry may be stored, and you can attach them to any entry regardless of that entry’s location in the DIT.

Note: Oracle Internet Directory does not enforce structure rules. It therefore handles both structural and auxiliary object classes in the same way.

Abstract Object Classes

An abstract object class is a virtual object class. It is used only for convenience when specifying the highest levels of the object class hierarchy. It cannot be the only object class for an entry. For example, the object class top is an abstract object class. It is required as a superclass for all structural object classes, but it cannot be used alone.

The top object class includes the mandatory attribute objectClass as well as several optional attributes. The optional attributes in top are:

- orclGuid: Global identification which remains constant if the entry is moved
- creatorsName: Name of the creator of the object class
- createTimeStamp: Time when the object class was created
- modifiersName: Name of the last person to modify the object class
- modifyTime: Time when the object class was last modified
- orclACI: access control list (ACL) directives that apply to all entries in the subtree below the access control policy point (ACP) where this attribute is defined
- orclEntryLevelACI: Access control policy pertaining to only a specific entity, for example, a special user
A directory naming context is a subtree that resides entirely on one server. It must be a complete subtree, that is, it must begin at an entry that serves as the top of the subtree, and extend downward to either leaf entries or references to subordinate naming contexts. It can range in size from a single entry to the entire directory information tree (DIT).

Figure 3–5 illustrates correct and incorrect naming contexts. Notice that the correct ones on the left are contiguous, and the incorrect ones on the right are not.

To enable users to discover specific naming contexts, you can publish those naming contexts in Oracle Internet Directory by using either Oracle Directory Manager or ldapmodify.

Security

Oracle Internet Directory is a key element of the Oracle Identity Management Infrastructure. This enables you to deploy multiple Oracle components to work against a shared instance of Oracle Internet Directory and associated infrastructure pieces. This sharing allows an enterprise to simplify security management across all applications.

In addition to the role it plays in the Oracle Identity Management infrastructure, Oracle Internet Directory provides many powerful features for protecting information.
These security features within Oracle Internet Directory itself include:

- Data integrity: Ensuring that data is not tampered with during transmission
- Data privacy: Ensuring that data is not inappropriately observed during transmission between Oracle Internet Directory and other components in the network.
- Authentication: Ensuring that the identities of users, hosts, and clients are correctly validated
- Authorization: Ensuring that a user reads or updates only the information for which that user has privileges
- Password policies: Establishing and enforcing rules for how passwords are defined and used
- Password protection: Ensuring that passwords are not easily discovered by others

You can use all these features to enforce a uniform security policy for multiple applications enabled for Oracle Internet Directory, and do so in either an enterprise or hosted environment. You do this by deploying the directory for administrative delegation. This deployment allows, for example, a global administrator to delegate to department administrators access to the metadata of applications in their departments. These department administrators can then control access to their department applications.

See Also:

- Chapter 16, "Directory Security Concepts" for a fuller discussion of the security features of Oracle Internet Directory
- Chapter 23, "Deployment of Oracle Identity Management Realms" for information on Oracle Internet Directory as it relates to the Oracle Identity Management infrastructure
- Chapter 21, "Delegation of Privileges for an Oracle Technology Deployment" for a discussion of how to protect applications in a large enterprise and in hosted environments
- The chapter on security in the Oracle Identity Management Integration Guide for a discussion of the unique aspects of security in an Oracle Directory Integration Platform environment
- Oracle Identity Management Infrastructure Administrator’s Guide for a fuller discussion of the Oracle Identity Management infrastructure

Globalization Support

Oracle Internet Directory follows LDAP Version 3 internationalization (I18N) standards. These standards require that the database storing directory data use Unicode Transformation Format 8-bit (UTF-8) character set. With Oracle9i, Oracle added a new UTF-8 character set called AL32UTF8. This database character set supports the latest version of Unicode (3.2), including the latest supplementary characters. This allows Oracle Internet Directory to store the character data of almost any language supported by Oracle Globalization Support. Moreover, although several different application program interfaces are involved in the Oracle Internet Directory implementation, Oracle Internet Directory ensures that the correct character encoding is used with each API.
Globalization Support means support for both single-byte and multibyte characters. A single-byte character is represented by one byte of memory. ASCII text, for example, uses single-byte characters. By contrast, a multibyte character can be represented by more than one byte. Simplified Chinese, for example, uses multibyte characters. An ASCII representation of a simplified Chinese directory entry definition might look like this:

dn: o=\274\327\271\307\316\304,c=\303\300\271\372
objectclass: top
objectclass: organization
o: \274\327\271\307\316\304

Where the attribute values correspond to an ASCII representation of a simplified Chinese directory entry definition.

By default, the main Oracle Internet Directory components—OID Monitor (OIDMON), OID Control Utility (OIDCTL), Oracle directory server (OIDLDAPD), Oracle directory replication server (OIDREPLD), and Oracle directory integration and provisioning server (ODISRV)—accept only the UTF-8 character set. The Oracle character set name is AL32UTF8.

The Oracle directory server and database tools are no longer restricted to run on a UTF8 database. However, be sure that all characters in the client character set are included in the database character set (with same or different character codes) if the database underlying the Oracle Internet Directory server is not AL32UTF8 or UTF8. Otherwise, there may be data loss during LDAP add, delete, modify, or modifydn operations if the client data cannot be mapped to the database character set.

Oracle Directory Manager, a Java-based tool, internally uses Unicode (UTF-16—that is, fixed-width 16-bit Unicode). It can support internationalized character sets.

See Also:

- "Oracle Internet Directory Architecture" on page 3-1 for information on the main Oracle Internet Directory components
- Appendix D, "Globalization Support in the Directory"

**Distributed Directories**

Although an online directory is logically centralized, it can be physically distributed onto several servers. This distribution reduces the work a single server would otherwise have to do, and enables the directory to accommodate a larger number of entries.

A distributed directory can be either replicated or partitioned. When information is replicated, the same naming contexts are stored by more than one server. When information is partitioned, one or more unique, non-overlapping naming contexts are stored on each directory server. In a distributed directory, some information may be partitioned and some may be replicated.

This section contains these topics:

- Directory Replication
- Directory Partitioning
Directory Replication

Replication is the process of copying and maintaining the same naming contexts on multiple directory servers. Some features of replication are:

- It improves performance by providing more servers to handle queries, and reliability by eliminating risks associated with a single point of failure.
- It can be either full or partial.
- Full replication involves propagating the entire DIT to another node.
- Partial replication involves propagating one or more subtrees, rather than the entire DIT, to another node.

Each copy of a naming context contained within a server is called a replica. Replicas can be read-only, updatable, or both. Servers that hold updatable replicas are called suppliers. Their changes are propagated to other servers called consumers.

The directory servers that participate in the replication of a given naming context form what is called a directory replication group (DRG). The relationship among the directory servers in a DRG is represented on each node by a special directory entry called a replication agreement. In a DRG, the protocol for transferring data between nodes can be based on either Oracle Database Advanced Replication or LDAP.

A DRG can be either single-master, multimaster, or fan-out.

- A single-master replication group has only one supplier replicating changes to one or more consumers. Only the supplier can be updated, and consumers are read-only.
- Multimaster replication, also called peer-to-peer or n-way replication, 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.
- A fan-out replication group, also called a point-to-point replication group, has a supplier replicating directly to a consumer. That consumer can then replicate to one or more other consumers. The replication can be either full or partial.

Figure 3–6 shows a replicated directory.
Partitioning, in which each directory server stores one or more unique, non-overlapping naming contexts, is another way of distributing directory information.

Figure 3–7 shows a partitioned directory in which some naming contexts reside on different servers.

**Note:** Although there are no Internet standards for directory replication yet, such standards are being developed by the IETF. Oracle Internet Directory replication adheres to the IETF standard proposal for representing directory change information in change logs. It can use standard LDAP as a transport for transmitting these change logs between Oracle Internet Directory replicas.

**See Also:** Chapter 29, "Oracle Internet Directory Replication Concepts" for a more detailed discussion of replication, including: Oracle Database Advanced Replication architecture, LDAP-based replication, change log purging, conflict resolution, and the replication process.
In Figure 3–7 on page 3-19, four naming contexts reside on Server A:
- $dc=acme, dc=com$
- $c=us, dc=acme, dc=com$
- $c=uk, dc=acme, dc=com$
- $c=au, dc=acme, dc=com$

Two naming contexts on Server A are replicated on Server B:
- $dc=acme, dc=com$
- $c=au, dc=acme, dc=com$

The directory uses one or more knowledge reference to locate information that is requested of Server B, but that resides on Server A. It passes this information to a client in the form of a referral.

Knowledge References and Referrals

A knowledge reference provides the names and addresses of the various naming contexts held in another partition. For example, in Figure 3–7 on page 3-19, Server B uses knowledge references to point to the $c=us$ and $c=uk$ naming contexts on Server A. When Server B is asked for information residing on Server A, it sends back one or more referrals to Server A. Clients can then use these referrals to contact Server A.

Typically, each directory server contains both superior and subordinate knowledge references. Superior knowledge references point upward in the DIT toward the root.
They tie the partitioned naming context to its parent. Subordinate knowledge references point downward in the DIT to other partitions.

For example, in Figure 3–8 on page 3-20, Server B holds four naming contexts, two of which are superior to the others. These two superior naming contexts use subordinate knowledge references to point to their subordinate naming contexts. Conversely, the naming context on Server A has an immediate superior residing on Server B. Server A therefore uses a superior knowledge reference to point to its parent on Server B.

![Figure 3–8 Using Knowledge References to Point to Naming Contexts](image)

Naming contexts that start at the top of the DIT obviously cannot have a knowledge reference to a superior naming context.

---

**Note:** There are presently no Internet standards for enforcing the validity of knowledge references, and Oracle Internet Directory does not do so. It is up to the administrator to ensure consistency among knowledge references within an enterprise network.

Oracle recommends that permission for managing knowledge reference entries be restricted, as is the case with any other privileged administrative function such as schema or access control.

There are two kinds of referrals:

- **Smart referrals**

  These are returned to the client when the knowledge reference entry is in the scope of the search. It points the client to the server that stores the requested information.

  For example, suppose that:

  - Server A holds the naming context, `ou=server development,c=us,o=acme`, and has a knowledge reference to Server B.
Server B holds the naming context ou=sales,c=us,o=acme. When a client sends a request to Server A for information in ou=sales,c=us,o=acme, Server A provides the user with a referral to Server B.

Default referrals

These are returned when the base object is not in the directory, and the operation is performed in a naming context on another server. A default referral typically sends the client to a server that has more detailed information about the directory partitioning arrangement.

For example, suppose that Server A holds:
- The naming context c=us,o=acme
- A knowledge reference to Server PQR that has more knowledge about the overall directory partitioning arrangement

Now suppose that a client requests information on c=uk,o=acme. When Server A finds that it does not have the c=uk,o=acme naming context, it provides the client with a referral to Server PQR. From there, the client can find the server holding the requested naming context.

See Also: "Managing Knowledge References and Referrals" on page 8-10

Oracle Delegated Administration Services and the Oracle Internet Directory Self-Service Console

Oracle Delegated Administration Services is a set of pre-defined, Web-based units for performing directory operations on behalf of a user. This set of services frees directory administrators from the routine tasks of directory management by enabling them to delegate specific functions to other administrators and to end users. It provides most of the functionality that directory-enabled applications require, such as creating a user entry, creating a group entry, searching for entries, changing user passwords, and other employee-specific data.

You can use Oracle Delegated Administration Services to develop your own tools for administering application data in the directory. Or you can use the Oracle Internet Directory Self-Service Console, a tool based on Oracle Delegated Administration Services that comes ready-to-use with Oracle Internet Directory. This console is used by several Oracle components to provide delegated administration.

See Also: Oracle Identity Management Guide to Delegated Administration

The Service Registry and Service to Service Authentication

The Service Registry and the Service to Service Authentication framework are Oracle Internet Directory features that facilitate integration between Oracle technology components that request services from one another. The Service Registry provides a place to store information, so that the components can discover each other. The Service to Service Authentication framework allows one component to authenticate to another and establishes trust among them.

The Service Registry is a container in Oracle Internet Directory (under cn=Services, Cn=OracleContext) where components store connectivity information, such as protocol, and other information, such as type of service. During installation, each OCS
component registers its information in the Registry. At run-time the components
discover information registered by other components. Service Registry objects are
stored in the Oracle Internet Directory DIT in a component-specific Services container
in the rootOracleContext.

Service to Service Authentication is a framework that allows one service to
authenticate to another and establish trusts among the services. At install time, each of
the client services is provisioned with a username and password in Oracle Internet
Directory. In addition, each target service defines an authorization role in Oracle
Internet Directory to control which components should it trust. When a component
requests services of another component, the requestor must authenticate to the target
service like any other client, using its own identity and credentials. The requesting
service must also be listed in the Target services Trusted Application group (Default
Group: contrasted Applications, counterpoise, cn=OracleContext). The requesting
service also must send the user's identity so that the target service can authenticate the
user as well. The data is sent securely, using either Digest authentication or the target
service's native secure authentication.

Oracle Directory Integration Platform

Oracle Directory Integration Platform enables an enterprise to integrate its
applications and other directories with Oracle Internet Directory. It provides all the
interfaces and infrastructure necessary to keep the data in Oracle Internet Directory
consistent with that in enterprise applications and connected directories. It also makes
it easier for third-party vendors and developers to develop and deploy their own
connectivity agents.

For example, an enterprise might want employee records in its HR database to be
synchronized with Oracle Internet Directory. In addition, the enterprise may deploy
certain LDAP-enabled applications (such as OracleAS Portal) that need to be notified
whenever changes are applied to Oracle Internet Directory.

Based on the nature of integration, Oracle Directory Integration Platform provides two
distinct services:

- The synchronization integration service, which keeps connected directories
  consistent with the central Oracle Internet Directory
- The provisioning integration service, which sends notifications to target
  applications to reflect changes made to entries of interest, such as users and
groups

See Also: Oracle Identity Management Integration Guide

Oracle Internet Directory and Identity Management

Identity management is the process of managing the complete security life cycle for
network entities in an organization. Because Oracle Internet Directory is a key element
of the Oracle Identity Management infrastructure, it enables you to simplify security
management across all applications. To do this, you deploy multiple Oracle
components against a shared instance of Oracle Internet Directory. Matching the
deployment of Oracle Internet Directory with the security needs of your enterprise
requires careful planning.

This section contains these topics:

- About Identity Management
- About the Oracle Identity Management Infrastructure
About Identity Management

Identity management usually refers to the management of an organization’s application users. Steps in their security life cycle include account creation, suspension, privilege modification, and account deletion. The managed entities may also include devices, processes, applications, or anything else that needs to interact in a networked environment. They may also include users outside of the organization, for example customers, trading partners, or Web services.

Identity management is important to IT deployments because it can reduce administrative costs while at the same time improving security.

The Oracle Identity Management infrastructure enables deployments to manage centrally and securely all enterprise identities and their access to various applications in the enterprise. Identity management comprises these tasks:

- Creating enterprise identities and managing shared properties of these identities through a single enterprise-wide console
- Creating groups of enterprise identities
- Provisioning these identities in various services available in the enterprise. This includes:
  - Account creation
  - Account suspension
  - Account deletion
- Managing policies associated with these identities. These include:
  - Authorization policies
  - Authentication Policies
  - Privileges delegated to existing identities

About the Oracle Identity Management Infrastructure

Oracle Identity Management is an integrated infrastructure that Oracle products rely on for distributed security. It is part of the infrastructure of the Oracle Application Server and for other Oracle products as well. Figure 3–9 on page 3-24 illustrates the components of the Oracle Identity Management infrastructure and how various Oracle and third-party products rely on it.
As shown in Figure 3–9, the Oracle Identity Management infrastructure includes the following components and capabilities:

- Oracle Internet Directory, a scalable, robust LDAP Version 3-compliant directory service implemented on the Oracle Database.
- Oracle Directory Integration Platform, which permits synchronization between Oracle Internet Directory and other directories and user repositories and automatic provisioning services for Oracle components and applications and, through standard interfaces, third-party applications.
- Oracle Delegated Administration Services, which provides trusted proxy-based administration of directory information by users and application administrators.
- Oracle Application Server Single Sign-On, which provides single sign-on access to Oracle and third party web applications.
- Oracle Application Server Certificate Authority, which generates and publishes X.509 V3 PKI certificates to support strong authentication methods.

While Oracle Identity Management is designed to provide an enterprise infrastructure for Oracle products, it can also serve as a general-purpose identity management solution for user-written and third-party enterprise applications. It provides a robust and scalable enterprise-wide identity management platform for third-party applications, hardware, and network operating systems. Custom applications can leverage Oracle Identity Management through a set of documented and supported services and APIs, for example:

- Oracle Internet Directory provides LDAP APIs for C, Java, and PL/SQL, and is compatible with other LDAP SDKs.
- Oracle Delegated Administration Services provide a core self-service console that may be customized to support third-party applications. In addition, it provides a number of services for building customized administration interfaces that manipulate directory data.
- The Oracle Directory Synchronization Service facilitates the development and deployment of custom solutions for synchronizing Oracle Internet Directory with third-party directories and other user repositories.
The Oracle Directory Provisioning Integration Service enables you to provision third-party applications and integrate the Oracle environment with other provisioning systems.

Oracle Application Server Single Sign-On provides APIs for developing and deploying partner applications that share a single sign-on session with other Oracle Web applications.

JAZN, Oracle’s implementation of the JAAS standard, enables applications developed for the Web using Oracle’s J2EE environment to leverage the Oracle Identity Management infrastructure for authentication and authorization.

In addition, Oracle works with third-party application vendors to ensure that their applications can leverage Oracle Identity Management out of the box.

**See Also:** Oracle Identity Management Infrastructure Administrator’s Guide for more information about the Oracle Identity Management infrastructure

### Identity Management Realms

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 simplifies the setting of the identity management policies.

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.

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.

### Default Identity Management 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.

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.

### Identity Management Policies

The Oracle Identity Management infrastructure supports a flexible set of management policies which comprise:

- Directory structure and naming policies that enable you to:
– Customize the directory structure in Oracle Internet Directory for your deployment.
– Specify where various identities are to be located and how they are uniquely identified.

- Authentication policies that enable you to specify authentication methods and protocols supported by the Oracle Identity Management infrastructure
- Identity management authorizations that enable you to control access to certain privileged services and delegate administration wherever necessary

**Note:** In Oracle Internet Directory Release 9.0.2, the equivalent term for "identity management realm" was "subscriber".

---

**Resource Information**

To fulfill the requests of users, some Oracle components gather data from various repositories and services. To gather the data, these components require the following information:

- Information specifying the type of resource from which the data is to be gathered. The type of resource could be, for example, an Oracle Database. This is called resource type information.
- Information for connecting and authenticating users to the resources. This is called resource access information.

This section contains these topics:

- Resource Type Information
- Resource Access Information
- Location of Resource Information in the DIT

**Resource Type Information**

Information about the resources that an application uses to service a user request is called resource type information. A resource type can be, for example, an Oracle Database or a Java Database Connectivity Pluggable Data Source. Resource type information includes such items as the class used to authenticate a user, the user identifier, and the password.

You specify resource type information by using the Oracle Internet Directory Self-Service Console.

**Resource Access Information**

Information for connecting and authenticating users to the databases is called resource access information. It is stored in an entry called a resource access descriptor (RAD) from which it can be retrieved and shared by various Oracle components.

For example, to service the request of a user for a sales report, Oracle Reports queries multiple databases. When it does this, it does the following:

1. Retrieves the necessary connect information from the RAD
2. Uses that information to connect to those databases and to authenticate the user requesting the data
Once it has done this, it compiles the report.

You specify resource access information by using the Oracle Internet Directory Self-Service Console. You can specify resource access information for each individual user or commonly for all users. In the latter case, all users connecting to a given application use, by default, the same information to connect to the necessary databases. Oracle recommends defining default resource access information whenever an application has its own integrated account management—for example, where each user is defined within the application itself by means of a unique single sign-on user name.

**Location of Resource Information in the DIT**

Figure 3–10 shows where resource information is located in the DIT.

*Figure 3–10 Placement of Resource Access and Resource Type Information in the DIT*

As Figure 3–10 shows, the resource access and resource type information is stored in the Oracle Context.

Resource access information for each user is stored in the `cn=User Extensions` node in the Oracle Context. In this example, the `cn=User Extensions` node contains resource access information for both the default user and for specific users. In the latter cases, the resource access information includes that needed for accessing both the Sales and the Bug databases.

Resource access information for each application is stored in the object identified by the application name—in this example, `cn=Oracle Application Server Reports Services, cn=Products,cn=Oracle`
Context, dc=us, dc=acme, dc=com. This is the user information specific to that product.

Resource type information is stored in the container cn=resource types, cn=common, cn=products, cn=Oracle Context.

See Also:

- The section about managing your own resource information in Oracle Identity Management Guide to Delegated Administration for instructions for an end user to specify resource access information

- The section on creating user entries by using the self-service console in Oracle Identity Management Guide to Delegated Administration for instructions for an administrator to specify resource access information when creating user entries

- The section on configuring default resource access information in Oracle Identity Management Guide to Delegated Administration for instructions for an administrator to define commonly used resources that all users automatically inherit

- The section on creating a new resource type in Oracle Identity Management Guide to Delegated Administration for instructions for an administrator to specify resource types

- "Plug-in Schema Elements" in Oracle Identity Management User Reference

- Oracle Application Server Reports Services Publishing Reports to the Web in Reports at http://www.oracle.com/technology/documentation
At the end of a successful installation, the OID Monitor, `oidmon`, and an instance of the Directory Server, `oidldapd`, are running.

**Notes:**

- You can run multiple instances of the directory server is on the same computer. For example, you can run one instance in SSL mode and another in non-SSL mode.
- If you restart the computer, you can restart Oracle Application Server Infrastructure by following the steps described in the section "Starting OracleAS Infrastructure" in *Oracle Application Server Administrator’s Guide*.
- The Oracle Internet Directory servers—that is, the directory server, the directory replication server, and the Oracle Directory Integration and Provisioning server daemons—can be started only by the operating system user who installed the Oracle Internet Directory software.

Before configuring and using Oracle Internet Directory, you must perform the tasks described in this chapter.

This section contains these topics:

- Task 1: Reset the Default Security Configuration
- Task 2: Reset the Default Password for the Database
- Task 3: Run the OID Database Statistics Collection Tool
- Tasks to Perform After Upgrading from Release 9.0.2
- Determining LDAP Port Assignment on UNIX and Linux

**Task 1: Reset the Default Security Configuration**

To meet the needs of your environment, you must customize the default security configuration. *Table 4–1* lists and describes the tasks you must perform to do this.
Task 1: Reset the Default Security Configuration

Table 4–1  Tasks to Reset the Default Security Configuration

<table>
<thead>
<tr>
<th>Task Area</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Protect the subSchemaSubEntry subentry and its children</td>
<td>Information about the directory is contained in the subentry subSchemaSubEntry and its children. Oracle recommends that you control access to these objects.</td>
</tr>
<tr>
<td>Establish access to entries</td>
<td>When you load directory entries, you are creating a hierarchy of directory entries. You must therefore establish:</td>
</tr>
<tr>
<td></td>
<td>■ Permissions to load entries into this hierarchy</td>
</tr>
<tr>
<td></td>
<td>■ Directory access for clients that need read, modify, and write access to directory entries</td>
</tr>
<tr>
<td>Modify default access policies</td>
<td>Oracle Internet Directory is installed with a default security configuration described in Chapter 21, “Delegation of Privileges for an Oracle Technology Deployment”. Before you begin using the directory, you can modify this default configuration to meet the needs of your environment and ensure that each user has the appropriate authorization.</td>
</tr>
<tr>
<td>Modify the default password policy</td>
<td>Password policies are sets of rules that govern how passwords are used. Oracle Internet Directory is installed with a default password policy that you can modify to meet the needs of your environment.</td>
</tr>
<tr>
<td>Modify the password of the super user</td>
<td>The super user has full access to directory information. The default user name of the super user is orcladmin; the default password is the Oracle Application Server Administrator password that was specified during installation. Modify this password immediately after installation.</td>
</tr>
<tr>
<td>Enable privacy mode for sensitive attributes</td>
<td>You must enable privacy mode to ensure that users cannot retrieve sensitive attributes in clear text. See “Privacy of Retrieved Sensitive Attributes” on page 16-2.</td>
</tr>
</tbody>
</table>

See Also:


- Chapter 16, "Directory Security Concepts" for an introduction to data integrity, data privacy, authentication, authorization, and password policies.

- Chapter 18, "Directory Access Control" for a detailed explanation of access control options and instructions for setting up security.


- Chapter 19, "Password Policies in Oracle Internet Directory" for an explanation of the default password policy.
Task 2: Reset the Default Password for the Database

Oracle Internet Directory uses a password when connecting to its designated Oracle database. The default for this password is the same as that specified during installation for the Oracle Application Server administrator (ias_admin). Change this default password by using the OID Database Password Utility.

See Also: The oidpasswd command-line tool reference in Oracle Identity Management User Reference for syntax and usage notes

Task 3: Run the OID Database Statistics Collection Tool

If you load data into the directory by any means other than the bulkload tool (bulkload), then you must run the OID Database Statistics Collection tool, $ORACLE_HOME/ldap/admin/oidstats.sql, after loading. This enables the Oracle Optimizer to choose an optimal plan for executing queries corresponding to LDAP operations. You can run OID Database Statistics Collection tool at any time without shutting down any of the OID daemons.

See Also: The oidstats.sql command-line reference in Oracle Identity Management User Reference

Tasks to Perform After Upgrading from Release 9.0.2

If you have upgraded Oracle Internet Directory from Release 9.0.2 to Release 10.1.2, perform the following tasks.

Set ACL Policy on Groups Container after Upgrade from Release 9.0.2

When upgrading Oracle Internet Directory from Release 9.0.2 to Release 10.1.2, the following ACL policy needs to be set on the groups container in the realm. The ACL policy should allow members of the group cn=Common Group Attributes, cn=groups, Oracle_Context_DN browse, search, and read access for private and public groups—that is, for groups where orclIsVisible is either not set or is set to TRUE or FALSE. This ACL is described in the Oracle Internet Directory Administrator’s Guide, in Chapter 17, in the section "Default Privileges for Reading Common Group Attributes".

The "Common Group Attributes" group is used by OracleAS Portal to query private and public groups. The ACI must to be added on the groups container. Change the Realm DN to the DN of the Realm and the DN of groups container in the realm to the appropriate group search base.

dn: DN of groups container in the realm
changetype: modify
add: orclaci
orclaci: access to entry filter={!orclisvisible=false} by group="cn=Common Group Attributes, cn=groups, cn=Oracle Context, Realm DN" (browse)
orclaci: access to attr=(*) filter={!orclisvisible=false} by group="cn=Common

Caution: Be careful when modifying the default ACLs in any Oracle Context. Doing so can disable the security of Oracle components in your environment. See component-specific documentation for details on whether you can safely modify the default ACLs in an Oracle Context.

See Also: The oidpasswd command-line tool reference in Oracle Identity Management User Reference for syntax and usage notes.

See Also: The oidstats.sql command-line reference in Oracle Identity Management User Reference.
Determining LDAP Port Assignment on UNIX and Linux

During installation of Oracle Application Server or third-party products, you might be prompted for an Oracle Internet Directory or LDAP port. To find the specific port number assigned to Oracle Internet Directory at installation, see the file $ORACLE_HOME/config/ias.properties. Look for the entries OIDport and OIDsslport.

The default port for enabling LDAP at Oracle Internet Directory installation time is 389. The Oracle Universal Installer always tries that port as its first choice. However, on many UNIX computers, /etc/services includes a line for LDAP reserving port 389. When that line is present, the Installer opts instead for a port number between 3060 to 3129, inclusive.

To confirm the port at which Oracle Internet Directory is running, simply run the ldapbind command-line tool, supplying either the host name and port number specified in the portlist.ini file or an alternative port specified during the Oracle Internet Directory installation.
This chapter introduces the various administration tools of Oracle Internet Directory. It discusses the online administration tool, called Oracle Directory Manager, and tells you how to launch it, navigate through it, and connect to directory servers with it. It also introduces the command-line tools for ldap, bulk, and catalog operations.

This chapter contains these topics:

- Using Oracle Identity Management Grid Control Plug-in
- Using Oracle Directory Manager
- Using Oracle Internet Directory Server Manageability
- Using Command-Line Tools

Directory administration is also aided by the Oracle Delegated Administration Services, a set of pre-defined, Web-based units for performing directory operations on behalf of a user. It frees directory administrators from the more routine directory management tasks by enabling them to delegate specific functions to other administrators and to end users. You can use it, for example, to enable end users to modify their personal profile information (including Oracle Application Server Single Sign-On passwords) without requiring the intervention of an administrator.

One tool, created by using Oracle Delegated Administration Services, is the Oracle Internet Directory Self-Service Console. This ready-to-use application provides a single graphical interface for delegated administrators and end users to manage data in the directory.

See Also: Oracle Identity Management Guide to Delegated Administration

Using Oracle Identity Management Grid Control Plug-in

The Oracle Identity Management Grid Control Plug-in is described in the Oracle Identity Management Infrastructure Administrator’s Guide.

Using Oracle Directory Manager

Oracle Directory Manager is a Java-based tool for administering Oracle Internet Directory. 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 Oracle Directory Manager

Before you can launch Oracle Directory Manager, you must have a directory server instance running. If no server instance is running, start one as described in "Semantics of OPMN Starting Oracle Internet Directory" on page 6-2.

See Also: "Oracle Internet Directory Architecture" on page 3-1 for a conceptual explanation of directory server instances

To start Oracle Directory Manager, follow the instructions for your operating system as described in Table 5–1:

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Windows</td>
<td>From the Start menu, choose Programs, then ORACLE_HOME, then Integrated Management, then Oracle Directory Manager</td>
</tr>
</tbody>
</table>
| UNIX             | If you have added $ORACLE_HOME/bin to $PATH, simply type the following at the system prompt: `oidadmin`  
|                  | If you have not added $ORACLE_HOME/bin to $PATH, navigate to $ORACLE_HOME/bin and enter: `./oidadmin` at the system prompt. |

The first time you start Oracle Directory Manager, 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 Oracle Directory Manager

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 Manager Connect dialog box appears.

2. In each field of the Credentials tab page, type the information specific to this server instance. These fields are described in Table A–1 on page A-2.
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. These fields are described in Table A–2 on page A-3.


**Navigating Oracle Directory Manager**

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

**Overview of Oracle Directory Manager**

Like the directory itself, the navigator pane (left side of the double window interface) has a tree-like structure. When Oracle Directory Manager first opens, the navigator pane shows only one tree item, Oracle Internet Directory Servers. 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, the changes you have made are committed, and the window remains available for more changes. If you press OK, the changes you have made are committed, and the window closes.

Similarly, some windows have buttons that are labeled Revert and Cancel. If you press Revert, then the changes you have made 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, the changes you have made in that window do not take effect, and the window closes.

**The Oracle Directory Manager Menu Bar**

Table 5–2 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.

---

**See Also:**

- Chapter 17, "Secure Sockets Layer (SSL) and the Directory" for instructions on enabling SSL
- "Entries" on page 3-7 for instructions on formatting distinguished names
- "Configuring SSL Parameters" on page 17-2 for information about changing ports and their impact on security
- *Oracle Advanced Security Administrator’s Guide* for instructions on creating a wallet by using Oracle Wallet Manager when using SSL
### Table 5–2  Oracle Directory Manager Menu Bar

<table>
<thead>
<tr>
<th>Menu</th>
<th>Menu Items</th>
</tr>
</thead>
</table>
| **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 Oracle Directory Manager. |
| **Edit** | **Edit**: Modifies an object.  
**Remove**: Removes a selected object.  
**Find Object Classes** or **Find Attributes**: Searches for either an object class or an attribute, depending on the context. If, in the navigator pane, you select Oracle Internet Directory, then directory server instance, then Server Management, then Object Classes, then this menu item searches for an object class. If you navigate to Oracle Internet Directory, then directory server instance, then Server Management, then Attributes, then it searches for attributes. |
| **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 Oracle Directory Manager's right pane. This is useful when comparing two pieces of information. |
| **Operation** | **Create Object Class**: Displays the New Object Class dialog box that you use to add a new object class.  
**Create Attribute**: Displays the New Attribute Type dialog box that you use to add a new attribute to an entry.  
**Create Access Cutler Point**: Displays the New Access Control Point dialog box that you use to add a new access control policy point (ACP).  
**Create Entry**: Displays the New Entry dialog box that you use to add a new directory entry.  
**Refresh Entry**: Updates data for entries stored in memory to reflect changes in the database.  
**Refresh Subtree Entries**: Updates the children of entries stored in memory to reflect changes in the database.  
**Configure Search Filter**: Narrows the range of entries the navigator pane displays according to whatever filter you specify.  
**Drop Index**: Removes an index from an attribute. When you select this item, an alert asks you to confirm that you want to drop the index.  
**Search**: Enables you to configure ACP searches.  
**User Preferences**: Displays a dialog box that enables you to:  
- Configure the display of entry search results  
- Establish whether ACPs are displayed whenever Oracle Directory Manager runs, or only as the result of a search. |
| **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. |
### The Oracle Directory Manager Toolbar

Table 5–3 illustrates and describes the buttons in the Oracle Internet Directory toolbar. Buttons become enabled or disabled depending on the pane or tab page you are displaying in Oracle Directory Manager.

<table>
<thead>
<tr>
<th>Button</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Connect/Disconnect" /></td>
<td>Connects to or disconnect from a directory server selected in the navigator pane.</td>
</tr>
<tr>
<td><img src="image" alt="Refresh" /></td>
<td>Updates data for objects other than entries that are stored in memory to reflect changes in the database.</td>
</tr>
<tr>
<td><img src="image" alt="Create" /></td>
<td>Adds a new object.</td>
</tr>
<tr>
<td><img src="image" alt="Create Like" /></td>
<td>Adds a new object by using another object as a template.</td>
</tr>
<tr>
<td><img src="image" alt="Edit" /></td>
<td>Modifies an object.</td>
</tr>
<tr>
<td><img src="image" alt="Find Object Classes or Attributes" /></td>
<td>Searches for either an object class or an attribute, depending on the context. If, in the navigator pane, you navigate to Oracle Internet Directory &gt; directory server instance &gt; Server Management &gt; Object Classes, then this button searches for an object class. If you navigate to Oracle Internet Directory &gt; directory server instance &gt; Server Management &gt; Attributes, then it searches for attributes.</td>
</tr>
<tr>
<td><img src="image" alt="Delete" /></td>
<td>Removes an object.</td>
</tr>
<tr>
<td><img src="image" alt="Add Object Classes" /></td>
<td>Adds an object class to an existing entry.</td>
</tr>
<tr>
<td><img src="image" alt="Refresh Entry" /></td>
<td>Updates data for entries stored in memory to reflect changes in the database.</td>
</tr>
<tr>
<td><img src="image" alt="Refresh Subtree Entries" /></td>
<td>Updates the children of entries stored in memory to reflect changes in the database.</td>
</tr>
<tr>
<td><img src="image" alt="Configure Search Filter" /></td>
<td>Narrows the range of entries the navigator pane displays according to whatever filter you specify.</td>
</tr>
<tr>
<td><img src="image" alt="Drop Index" /></td>
<td>Removes an index from an attribute. When you click this button, an alert asks you to confirm that you want to drop the index.</td>
</tr>
<tr>
<td><img src="image" alt="Search" /></td>
<td>Enables you to configure ACP searches.</td>
</tr>
<tr>
<td><img src="image" alt="User Preferences" /></td>
<td>Enables you to configure the display of ACPs in the navigator pane, as well as entries in a search operation.</td>
</tr>
<tr>
<td><img src="image" alt="Help" /></td>
<td>Displays the Help system.</td>
</tr>
</tbody>
</table>

### Connecting to Additional Directory Servers by Using Oracle Directory Manager

You can connect to more than one directory server at a time, and then view and modify the data, schema, and security for each directory server. If you do this, then each server is listed in the navigator pane under Oracle Internet Directory Servers.

To connect to an additional directory server:

1. In the navigator pane, select Oracle Internet Directory Servers.
2. In the right pane, choose **New**.

3. Follow the login procedures described earlier in this chapter, in "Connecting to a Directory Server by Using Oracle Directory Manager" on page 5-2.

**Disconnecting from a Directory Server by Using Oracle Directory Manager**

To disconnect from a directory server by using Oracle Directory Manager, from the **File** menu choose **Disconnect**. Also, when you exit Oracle Directory Manager, 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 Oracle Directory Manager, all previously connected server connections appear in the Directory Server Login dialog box.

**Configuring the Display and Duration of Searches in Oracle Directory Manager**

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

If you make the configuration in both Oracle Directory Manager and the directory server, and the configuration in Oracle Directory Manager does not match the one in the directory server, then Oracle Internet Directory resolves the conflict as follows:

- If the value you set in Oracle Directory Manager is greater than that in the directory server, then the configuration of the server prevails. For example, if you set Oracle Directory Manager 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 Oracle Directory Manager is less than that in the directory server, then the configuration of Oracle Directory Manager prevails. For example, if you set Oracle Directory Manager 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 Oracle Directory Manager:

1. In the navigator pane, expand **Oracle Internet Directory Servers**, and 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 **Maximum number of one-level subtree entries** field, enter the maximum number of entries to be returned by a search.

4. In the **Search Time Limit** field, enter the maximum number of seconds for a search to be completed. The default is 3600.

5. Choose **OK**.

To configure the display and duration of searches in an Oracle directory server:

1. In the navigator pane, expand **Oracle Internet Directory Servers** and select a directory server instance. The group of tab pages for that server appear in the right pane.
2. In the **System Operational Attributes** tab page, in the **Query Entry Return Limit** field, enter the maximum number of entries to be returned by a search. The default is 1000.

3. In the **Server Operation Time Limit** field, enter the maximum number of seconds for a search to be completed. The default is 3600.

4. Choose **Apply**.

**Performing Administrative Tasks by Using Oracle Directory Manager**

You can perform most of the Oracle Internet Directory administrative tasks through Oracle Directory Manager. Those that you cannot perform through Oracle Directory Manager involve running processes, such as starting and stopping the OID Monitor (oidmon) and starting and stopping server instances. To perform tasks that you cannot perform with Oracle Directory Manager, use the appropriate LDAP command-line tool.

See Also:

- "Using Command-Line Tools" on page 5-8
- "Oracle Identity Management Command-line Tool Reference" in *Oracle Identity Management User Reference*

Table 5–4 lists the task areas you can manage by using Oracle Directory Manager and where to find instructions for each area.

<table>
<thead>
<tr>
<th>Task Area</th>
<th>Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access Control Management</td>
<td>&quot;Managing Access Control by Using Oracle Directory Manager&quot; on page 18-14</td>
</tr>
<tr>
<td></td>
<td>&quot;Managing Access Control by Using Command-Line Tools&quot; on page 18-35</td>
</tr>
<tr>
<td>Attribute Uniqueness</td>
<td>Chapter 10, &quot;Attribute Uniqueness in the Directory&quot;</td>
</tr>
<tr>
<td>Management</td>
<td></td>
</tr>
<tr>
<td>Audit Log Management</td>
<td>Chapter 14, &quot;Logging, Auditing, and Monitoring the Directory&quot;</td>
</tr>
<tr>
<td>Change Log</td>
<td>&quot;Change Logs in Directory Replication&quot; on page 29-19</td>
</tr>
<tr>
<td>Management</td>
<td>Chapter 30, &quot;Oracle Internet Directory Replication Installation and</td>
</tr>
<tr>
<td></td>
<td>Configuration&quot;</td>
</tr>
<tr>
<td></td>
<td>The chapter on the Oracle Directory Synchronization Service in</td>
</tr>
<tr>
<td></td>
<td>Oracle Identity Management Integration Guide</td>
</tr>
<tr>
<td></td>
<td>The chapter on the Oracle directory integration and provisioning server in</td>
</tr>
<tr>
<td></td>
<td>Oracle Identity Management Integration Guide</td>
</tr>
<tr>
<td>Entry Management</td>
<td>&quot;Managing Entries by Using Oracle Directory Manager&quot; on page 8-1</td>
</tr>
<tr>
<td>Garbage Collection</td>
<td>Chapter 26, &quot;Garbage Collection in Oracle Internet Directory&quot;</td>
</tr>
<tr>
<td>Management</td>
<td></td>
</tr>
<tr>
<td>Password Policy Management</td>
<td>Chapter 19, &quot;Password Policies in Oracle Internet Directory&quot;</td>
</tr>
<tr>
<td>Password Verifier</td>
<td>Chapter 20, &quot;Directory Storage of Password Verifiers&quot;</td>
</tr>
<tr>
<td>Management</td>
<td></td>
</tr>
<tr>
<td>Plug-in Management</td>
<td>Part VIII, &quot;Directory Plug-ins&quot;</td>
</tr>
</tbody>
</table>
Using Oracle Internet Directory Server Manageability

Oracle Internet Directory Server Manageability enables you to monitor various types of information about Oracle Internet Directory servers. You can also use Oracle Internet Directory Server Manageability to start, stop, or restart a directory server instance. To exploit the features of Oracle Internet Directory Server Manageability, you use Oracle Enterprise Manager 10g Application Server Control Console.

See Also: Monitoring Oracle Internet Directory Servers

Using Command-Line Tools

Oracle Internet Directory provides several types of command-line tools for manipulating directory entries and attributes—for example:

- LDAP tools, for altering objects in text files written in the LDAP Data Interchange Format (LDIF)
- A catalog management tool, for indexing existing attributes
- Various tools to help you synchronize multiple directories in your enterprise

Many of the command-line tools act on objects that are in text files written in the LDAP Data Interchange Format (LDIF).

Note: To use the command-line tools, set the following environment variables:

- ORACLE_HOME.
- ORACLE_SID or a proper TNS CONNECT string.
- NLS_LANG (APPROPRIATE_LANGUAGE,AL32UTF8). The default language set at installation is AMERICAN_AMERICA.
- PATH and CLASSPATH. In the PATH and CLASSPATH environment variables, specify the Oracle LDAP binary—that is, ORACLE_HOME/bin—before the UNIX binary directory.

Note: Command-line examples in Oracle Identity Management documentation are based on the UNIX ksh. Arguments that must be escaped from the shell are shown in double quotes ("'). Use the appropriate quote characters for your shell environment.
This section contains these topics:

- Command-Line Tools for Starting, Stopping, and Monitoring Oracle Internet Directory Servers
- Command-Line Tools for Managing Entries and Attributes
- Command-Line Tools for Performing Bulk Operations
- Command-Line Tools for Managing Replication
- OID Migration Tool (ldifmigrator)
- OID Database Statistics Tool (oidstats.sql)
- OID Database Password Utility (oidpasswd)

Command-Line Tools for Starting, Stopping, and Monitoring Oracle Internet Directory Servers

Table 5–5 lists and describes the various command-line tools for starting, stopping, and monitoring Oracle Internet Directory servers and points you to more information about each one.

See Also: Chapter 6, "Process Control of Oracle Internet Directory Components"

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
</table>
| Oracle Process Manager and Notification Server (OPMN) | Use OPMNCTL to stop or start Oracle Internet Directory as a component of Oracle Application Server. | The opmnctl command-line tool reference in Oracle Identity Management User Reference  
The chapter entitled "Configuring Oracle Internet Directory" in Oracle Process Manager and Notification Server Administrator's Guide |
| OID Control Utility (OIDCTL)            | Use this tool to start and stop an individual instance of the server. The commands are interpreted and executed by the OID Monitor process. | "Oracle Internet Directory Architecture" on page 3-1 for a conceptual description  
The oidctl command-line tool reference in Oracle Identity Management User Reference |
Command-Line Tools for Managing Entries and Attributes

Table 5–6 lists and describes the command-line tools for managing entries and attributes, and points you to further information.

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Catalog Management Tool</td>
<td>Oracle Internet Directory uses indexes to make attributes available for</td>
<td>The catalog command-line tool reference in Oracle Identity Management User Reference for syntax and usage notes</td>
</tr>
<tr>
<td>(catalog)</td>
<td>searches. When Oracle Internet Directory is installed, the entry cn=catalogs lists available attributes that can be used in a search. Only those attributes that have an equality matching rule can be indexed. If you want to use additional attributes in search filters, you must add them to the catalog entry. You can do this at the time you create the attribute by using Oracle Directory Manager. However, if the attribute already exists, then you can index it only by using the Catalog Management tool. Useful in creating and dropping the indexes.</td>
<td></td>
</tr>
<tr>
<td>ldapadd</td>
<td>Use this tool to add entries one at a time.</td>
<td>The ldapadd command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>ldapaddmt</td>
<td>Use this tool to add several entries concurrently by using this shared-server tool.</td>
<td>The ldapaddmt command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>ldapbind</td>
<td>Use this tool to authenticate user/client to a directory server.</td>
<td>The ldapbind command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>ldapcompare</td>
<td>Use this tool to see whether an entry contains a specified attribute value.</td>
<td>The ldapcompare command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>ldapdelete</td>
<td>Use this tool to delete entries.</td>
<td>The ldapdelete command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>ldapmoddn</td>
<td>Use this tool to modify the DN or RDN of an entry, rename an entry or a subtree, or move an entry or a subtree under a new parent.</td>
<td>The ldapmoddn command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>ldapmodify</td>
<td>Use this tool to create, update, and delete attribute data for an entry.</td>
<td>The ldapmodify command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>ldapmodifymt</td>
<td>Use this tool to modify several entries concurrently by using this shared-server tool.</td>
<td>The ldapmodifymt command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>ldapsearch</td>
<td>Use this tool to search for directory entries.</td>
<td>The ldapsearch command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
</tbody>
</table>

Command-Line Tools for Performing Bulk Operations

Table 5–7 lists and describes the command-line tools for performing bulk operations, and points you to further information.
Table 5–7  Command-Line Tools for Performing Bulk Operations

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
<th>More Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>bulkdelete</td>
<td>Use this tool to delete a subtree efficiently</td>
<td>The bulkdelete command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>bulkload</td>
<td>Use this tool to load and append large numbers of entries to Oracle Internet Directory through LDIF files</td>
<td>The bulkload command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>bulkmodify</td>
<td>Use this tool to modify a large number of existing entries efficiently</td>
<td>The bulkmodify command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>ldifwrite</td>
<td>Use this tool to copy data from the directory information base into an LDIF file that can be read by any LDAP-compliant directory server. You can use ldifwrite in conjunction with bulkload. You can also use ldifwrite to back up information from all or part of a directory.</td>
<td>The ldifwrite command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
</tbody>
</table>

Command-Line Tools for Managing Replication

Table 5–8 on page 5-12 lists and describes the command-line tools for managing replication, and points you to further information.
Replication Environment Management Tool

This tool ensures that Advanced Replication is properly configured for directory replication. In the event of a directory replication failure, this tool looks for the problems and seeks to rectify them. If it cannot solve the problem, then it gives you a report of the nature of the problem and points you to a possible solution.

Oracle Internet Directory Comparison and Reconciliation Tool

When a replication conflict arises, Oracle directory replication server places the change in the retry queue and tries to apply it from there for a specified number of times. If it fails after that specified number, then the replication server puts the change in the human intervention queue. From there, the replication server repeats the change application process at less frequent intervals while awaiting your action.

At this point, you need to:
1. Examine the change in the human intervention queues
2. Reconcile the conflicting changes on the consumer with those on the supplier by using the Oracle Internet Directory Comparison and Reconciliation Tool
3. Place the change either back into the retry queue or into the purge queue

Human Intervention Queue Manipulation Tool

Once you have reconciled conflicting changes by using the Oracle Internet Directory Comparison and Reconciliation Tool, the Human Intervention Queue Manipulation Tool enables you to move them from the human intervention queue to either the retry queue or the purge queue. Moving the change to the purge queue means that there are no further attempts to re-apply the change log entry.

OID Migration Tool (ldifmigrator)

Use this tool to migrate data from application-specific repositories into Oracle Internet Directory.

See Also: The ldifmigrator command-line tool reference in Oracle Identity Management User Reference for instructions on using this tool

OID Database Statistics Tool (oidstats.sql)

Use this tool to analyze the various database ods schema objects to estimate the statistics. You must run this utility whenever there are significant changes in directory data—including the initial load of data into the directory.

If you load data into the directory by any means other than the bulkload tool (bulkload), then you must run the OID Database Statistics Collection tool after loading. Statistics collection is essential for the Oracle Optimizer to choose an optimal plan in executing the queries corresponding to the LDAP operations. You can run OID Database Statistics Collection tool at any time, without shutting down any of the OID daemons.
OID Database Password Utility (oidpasswd)

The OID Database Password Utility is used to:

- Change the password to the Oracle Internet Directory database.
  
  Oracle Internet Directory uses a password when connecting to an Oracle database. The default for this password matches the value you specified during installation for the Oracle Application Server administrator’s password. You can change this password by using the OID Database Password Utility.

- Create a wallet, named `oidpwd1ldap1`, for the Oracle Internet Directory database password, and a wallet, named `oidpwdrsid`, for the Oracle directory replication server password.
  
  The `sid` is obtained not from the environment variable `SID` but from the connected database.

  With the `create_wallet=true` option, you need to provide the ODS password to authenticate yourself to the ODS database before the ODS wallet can be generated. Note that the default ODS password is the same as that for the Oracle Application Server administrator.

- Unlock a locked directory super user account, namely, `cn=orcladmin`.

**Note:** To change the ODS database user password, you must use the `oidpasswd` tool. If you change the ODS database user password by any other means, then Oracle Internet Directory instances fail to start.

- Reset the super user password.

- Manage super user restricted ACPs.

**See Also:** The `oidstats.sql` command-line tool reference in Oracle Identity Management User Reference

**See Also:** The `oidpasswd` command-line tool reference in Oracle Identity Management User Reference
This chapter describes the concepts behind the process control model in Oracle Internet Directory. The process control model applies to the Oracle Internet Directory LDAP server, the Replication server, and the Directory Integration Server.

This section contains these topics:

- Tools and Daemons Important to Oracle Internet Directory Process Control
- Oracle Internet Directory Integration with OPMN
- Oracle Internet Directory Process Control—Best Practices
- OIDMON and the ODS_PROCESS Table
- OIDCTL Process Control Semantics

For information on starting and stopping the Oracle Directory Integration Platform server, see the chapter on managing the Oracle Directory Integration Platform server in Oracle Identity Management Integration Guide.

Tools and Daemons Important to Oracle Internet Directory Process Control

The Oracle Process Manager and Notification Server (OPMN) is a daemon process that monitors the components in an installation of Oracle Application Server. It is installed and configured in every middle-tier and Oracle Application Server Infrastructure installation and is essential for running Oracle Application Server. Because Oracle Internet Directory is part of the Oracle Application Server Infrastructure, OPMN is responsible for monitoring Oracle Internet Directory as an Oracle Application Server component. The command-line interface to OPMN is $ORACLE_HOME/opmn/bin/opmnctl. You use OPMNCTL to stop or start Oracle Internet Directory as a component.

See Also: The chapter entitled "Configuring Oracle Internet Directory" in Oracle Process Manager and Notification Server Administrator's Guide

OIDMON ($ORACLE_HOME/bin/oidmon) is a daemon process responsible for the process control of all Oracle Internet Directory Server instances.

OIDCTL ($ORACLE_HOME/bin/oidctl) is a command line tool that lets you configure additional Oracle Internet Directory Server Instances or perform process control at the instance level.
Oracle Internet Directory Integration with OPMN

This section describes Oracle Internet Directory interaction with OPMN. It includes the following sections:

- Semantics of OPMN Monitoring Oracle Internet Directory
- Oracle Internet Directory Snippet in OPMN.XML
- Semantics of OPMN Starting Oracle Internet Directory
- Semantics of OPMN Stopping Oracle Internet Directory
- Semantics of OPMN Monitoring OIDMON

Semantics of OPMN Monitoring Oracle Internet Directory

Monitoring rules are as follows:

- OPMN is responsible for monitoring Oracle Internet Directory as an Oracle Application Server component.
- OPMN integration with Oracle Internet Directory is such that OPMN knows only about OIDMON and is unaware of the Oracle Internet Directory Server Instances.
- OPMN is responsible for the direct start, stop, restart and monitoring of OIDMON only. OIDMON continues to be responsible for the direct start, stop, restart and monitoring of all Oracle Internet Directory Server Instances.

Oracle Internet Directory Snippet in OPMN.XML

Oracle Internet Directory component-specific directives are located as follows:

- Oracle Internet Directory component-specific directives for OPMN are located under the tag `<ias-component id="OID" status="enabled">` in `$ORACLE_HOME/opmn/conf/opmn.xml`.
- OPMN uses the directives in the OID component snippet in `opmn.xml` and invokes OIDMON and OIDCTL as required.
- OIDCTL related requirements are located under the tag `<category id="oidctl parameters">`.
- OIDMON related requirements are located under the tag `<category id="oidmon parameters">`. There should be only one such directive.
- The default value of OID Snippet in `opmn.xml` has one entry for OIDMON and one for OIDCTL.

See Also: The chapter entitled "Configuring Oracle Internet Directory" in *Oracle Process Manager and Notification Server Administrator's Guide*

Semantics of OPMN Starting Oracle Internet Directory

OPMN startup of an Oracle Internet Directory component proceeds as follows:

- You can indicate to OPMN the intent to start an Oracle Internet Directory component with one of the following commands:

  ```bash
  opmnctl startall
  opmnctl startproc ias-component=OID
  ```
OPMN issues an `oidmon start` command with appropriate arguments to OIDMON as specified in the contents of "oidmon parameters" in the OID Snippet in `opmn.xml`.

OPMN issues `oidctl start` commands if the OID Snippet in `opmn.xml` has entries that require this.

Note that when you use `opmnctl startproc ias-component=OID`, the `opmn.xml` parameters are not reloaded as they are when you use `opmnctl startall`.

Semantics of OPMN Stopping Oracle Internet Directory

OPMN stopping of an Oracle Internet Directory component proceeds as follows:

- You can indicate to OPMN the intent to stop an Oracle Internet Directory component with one of the following commands:
  - `opmnctl stopall`
  - `opmnctl stopproc ias-component=OID`

- OPMN issues an `oidmon stop`.

- OPMN does not issue `oidctl stop` commands; instead, the OIDMON stop semantics ensure that the Oracle Internet Directory Server Instances are stopped as required. For more information, see "OIDMON and the ODS_PROCESS Table" on page 6-6.

Note that when you use `opmnctl stopproc ias-component=OID`, the `opmn.xml` parameters are not reloaded as they are when you use `opmnctl stopall`.

Semantics of OPMN Monitoring OIDMON

OPMN monitors OIDMON as follows:

- Once you start OIDMON through OPMN, OPMN ensures that OIDMON is up and running. If OIDMON goes down for some reason, OPMN will bring it back up.

- If you issue `oidmon stop` on the command line, OIDMON will be stopped but OPMN will immediately bring it back up.

Oracle Internet Directory Process Control–Best Practices

The recommended approach for using `OPMNCTL` and `OIDCTL` is as follows:

- Use OPMNCTL to stop or start Oracle Internet Directory as a component. That is, use it to stop or start all Oracle Internet Directory LDAP, replication, and Oracle Directory Integration Platform server instances.
  - Using OPMNCTL to stop Oracle Internet Directory causes OPMN to issue an `oidmon stop`, which results in OIDMON shutting down all configured LDAP, replication, and Oracle Directory Integration Platform server instances.
  - Using OPMNCTL to start Oracle Internet Directory causes OPMN to issue an `oidmon start`, which results in OIDMON starting up all configured LDAP, replication, and Oracle Directory Integration Platform server instances.
Use OIDCTL to configure required additional Oracle Internet Directory Server Instances.
- To configure an Oracle Internet Directory LDAP, replication, or Oracle Directory Integration Platform server that is not part of the default configuration, use the OIDCTL command to start such an instance.
- Issue the `oidctl start` command only once for each instance for the duration of the deployment of that configuration. The OIDMON start and stop semantics ensure that configured servers start and stop appropriately.

Use OIDCTL to perform process control at the instance level only.
- Use `oidctl stop` to stop a particular instance of the Oracle Internet Directory LDAP, replication, or Oracle Directory Integration Platform server. Do not use it to stop all the configured instances of the servers.
- Use `oidctl start` to start an additional instance of the Oracle Internet Directory LDAP, replication, or Oracle Directory Integration Platform server that is not already configured.
- If, for some reason, a server instance is stopped using `oidctl stop`, then it is removed from the process table and will not be known to OIDMON any longer. To restart an instance that was stopped this way, use `oidctl start`.

The following sections provide examples of the recommended approach. The examples are:

- Changing the Configuration of the Default OID LDAP Server Instance
- Configuring Additional Oracle Internet Directory LDAP Server Instances
- Deconfiguring the Default Oracle Internet Directory LDAP Server Instance
- Configuring an Instance of the Oracle Internet Directory Replication Server
- Configuring an Oracle Directory Integration Platform Server Instance

See Also: "Oracle Internet Directory Server Administration Tools" in Oracle Identity Management User Reference for more information on the syntax of the commands used in the examples

### Changing the Configuration of the Default OID LDAP Server Instance

A default Oracle Internet Directory installation uses the default configuration set (configset0), which provides a single server instance with one server process and two database connections configured. This configuration might be inadequate to handle the production LDAP load in your environment. If so, you need to increase the number of server processes or database connections or both. You change these by changing the `orclserverprocs` and `orclmaxcc` attributes values, respectively, in configset0.

You change the default configset using Oracle Directory Manager, as follows:

1. Launch Oracle Directory Manager.
2. Log in as `orcladmin`.
3. Expand Server Management.
4. Click Default Configuration.
5. Change Max Number of DB Connections to the desired value. A typical recommendation is 10.
6. Change **Number of Child Process** if required. A typical value is 1 or the number of CPUs on the system.

7. Click **Apply**.

8. Restart the Oracle Internet Directory servers, as follows:

   opmnctl stopproc ias-component=OID
   opmnctl startproc ias-component=OID

Oracle recommends that you not change other parameters in the default configuration set.

### Configuring Additional Oracle Internet Directory LDAP Server Instances

To start additional Oracle Internet Directory LDAP server instances, add additional configuration sets with the required configuration values and use these additional configuration sets to start additional server instances. (Do not use the default configuration set and override the default values.) You add a configuration set and to use it to start an Oracle Internet Directory LDAP Server Instance as follows:

1. Launch Oracle Directory Manager.

2. Expand **Server Management**.

3. Expand **Directory Server**.

4. Right click **Default Configuration Set**.

5. Click **Create Like**.

6. Change the required parameters in the new configuration set. Ensure that the port numbers do not overlap with those of the default configuration set or any other configuration set.

7. Click **OK**.

   To start an LDAP server instance using a new configuration set called **configset2**, type:

   oidctl connect=connStr server=oidldapd instance=2 configset=2 start

### Deconfiguring the Default Oracle Internet Directory LDAP Server Instance

To replace the Oracle Internet Directory LDAP server instance with one or more Oracle Internet Directory LDAP Server Instances, you must edit `opmn.xml` to deconfigure the default LDAP instance. By default, `opmn.xml` contains an XML snippet that attempts to start the default Oracle Internet Directory LDAP server instance when you type `opmnctl start`. To deconfigure the default Oracle Internet Directory LDAP server instance, perform the following steps:

1. Type:

   oidctl connect=connStr server=oidldapd instance=1 stop

2. Edit the file `$ORACLE_HOME/opmn/conf/opmn.xml` and remove the following lines:

   ```xml
   <category id="oidctl-parameters">
   <data id="connect" value="iasdb"/>
   <data id="startoidldapd" value="true"/>
   </category>
   ```
Configuring an Instance of the Oracle Internet Directory Replication Server

To configure an instance of OID Replication Server, use the `oidctl start` command. For example:

```
oidctl connect=connStr server=oidrepld instance=1 \flags="-h LdapHost -p LdapPort" start
```

Do not start more than one instance of `oidrepld`.

Configuring an Oracle Directory Integration Platform Server Instance

To configure an instance of the Oracle Directory Integration Platform Server, use the `oidctl start` command. For example:

```
oidctl connect=connStr server=odisrv instance=1 \flags="-h LdapHost -p LdapPort" start
```

OIDMON and the ODS_PROCESS Table

OIDMON It is responsible for starting, stopping, restarting, and monitoring of all the Oracle Internet Directory Server instances including Oracle Internet Directory LDAP, Replication, and Directory Server instances.

OIDMON reads the contents of the ods_process table in the ODS database user schema once every periodicity and acts upon the intent conveyed by the contents of that table. The periodicity is controlled by the value of the `sleep` command line argument used at oidmon startup, and the default value is 10 seconds.

Table 6–1 describes the information in the ODS_PROCESS table that is relevant to process control:

<table>
<thead>
<tr>
<th>Item</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Instance</td>
<td>Unique instance number for a given server ID on a given host</td>
</tr>
<tr>
<td>PID</td>
<td>Process ID of the server that is up and running</td>
</tr>
<tr>
<td>ServerID</td>
<td>Server ID (2=OIDLDAPD, 3=OIDREPLD, 7=ODISRV)</td>
</tr>
<tr>
<td>Flags</td>
<td>Command line arguments that need to be passed to the server instance</td>
</tr>
<tr>
<td>Hostname</td>
<td>Name of the host on which this server must be present</td>
</tr>
<tr>
<td>Configset</td>
<td>Configset information</td>
</tr>
<tr>
<td>State</td>
<td>State of the Server Instance (0=stop, 1=start, 2=running, 3=restart, 4=shutdown)</td>
</tr>
<tr>
<td>RetryCount</td>
<td>Number of attempts to start the server instance before it could be started successfully</td>
</tr>
</tbody>
</table>

Notes:

- There is a uniqueness constraint on (Instance,ServerID,Hostname).
- Details are provided here about ods_process only to convey the concepts. Any updates to the table by a user, other than by using OIDCTL, are inappropriate and are not supported by Oracle.
OIDMON takes the following actions with respect to Oracle Internet Directory server instances:

- When OIDMON is stopped, it performs the following tasks before shutting down:
  - OIDMON stops all the server instances active (up and running) on its node, that is, all instances active on the same host as OIDMON.
  - OIDMON updates the value of the "state" column of rows in the ods_process table with matching hostname to 4.

- When OIDMON is started, it starts all Oracle Internet Directory Server instances whose information in the ods_process table has state value 1 or 4 and hostname value matching the host on which OIDMON is active.

**OIDCTL Process Control Semantics**

This section explains the semantics of starting and stopping Oracle Internet Directory server instances by using OIDCTL.

OIDCTL communicates the intent to start, stop, or restart a particular Oracle Internet Directory Server instance by updating the ods_process table in the ODS database user schema.

When you start an Oracle Internet Directory server instance:

- You use the OIDCTL command line utility to indicate the intent to start a particular Oracle Internet Directory server instance.
- OIDCTL inserts a row into the ods_process table to convey the intent to OIDMON.
- If the uniqueness constraint on the ods_process table is violated, then OIDCTL reports the error "Instance number already in use".
- OIDMON reads this information, starts the server instance, and updates the state and PID columns of the ods_process table as appropriate.

When you stop an Oracle Internet Directory server instance by using OIDCTL:

- You use the OIDCTL command line utility to indicate the intent to stop a particular Oracle Internet Directory server instance.
- OIDCTL updates the corresponding row in the ods_process table to convey the intent to OIDMON.
- If the corresponding row is not found, that is, the given instance is not configured, then OIDCTL reports the error "Cannot Stop an Instance that is not running".
- OIDCTL updates the state value to 0.
- OIDMON reads this information, stops the server instance, and deletes the row representing this server instance from the ods_process table.
This part guides you through the tasks to configure and maintain Oracle Internet Directory. This part contains these chapters:

- Chapter 7, "Oracle Directory Server Administration"
- Chapter 8, "Directory Entries Administration"
- Chapter 9, "Using Bulk Tools"
- Chapter 10, "Attribute Uniqueness in the Directory"
- Chapter 11, "Directory Schema Administration"
- Chapter 12, "Referential Integrity"
- Chapter 13, "Dynamic and Static Groups in Oracle Internet Directory"
- Chapter 14, "Logging, Auditing, and Monitoring the Directory"
- Chapter 15, "Backup and Restoration of a Directory"
This chapter explains how to manage an Oracle directory server by using Oracle Directory Manager and command-line tools.

This chapter contains these topics:

- Managing Server Configuration Set Entries
- Setting System Operational Attributes
- Managing Naming Contexts
- Managing Super Users, Guest Users, and Proxy Users
- Managing Anonymous Binds
- Viewing Active Server Instance Information
- Closing Idle LDAP Connections
- Changing the Password to the Oracle Internet Directory Database Server
- Dereferencing Alias Entries
- Locating Directory Servers in a Distributed Environment

See Also: Chapter 4, "Post-Installation Tasks and Information" for instructions on starting and stopping directory server instances

Managing Server Configuration Set Entries

When you start an Oracle directory server by using the object class, that start message refers to a configuration set entry containing server parameters. You can add, modify, and delete configuration set entries by using either Oracle Directory Manager or the appropriate command-line tool.

This section contains these topics:

- Preliminary Considerations for Managing Configuration Set Entries
- Managing Server Configuration Set Entries by Using Oracle Directory Manager
- Managing Server Configuration Set Entries by Using Command-Line Tools

See Also: "Configuration Set Entries" on page 3-6 for a conceptual overview of configuration set entries
Preliminary Considerations for Managing Configuration Set Entries

The configuration set entry `configset0` is the default, and is used as the template for all new configuration set entries. Although you can change values in the default configuration set, all of your changes are then carried over to every new configuration set entry that you create.

To change values that should not be in effect for every server instance, it is better to create new configuration set entries. Note that this applies to the Oracle directory server and Oracle directory integration server instances only. The Oracle replication directory server supports only one configuration set.

You may want to establish a separate instance of a directory server with different values. If you do not want those values to be exercised by all users, then set up a new configuration set entry and run a separate server instance pointing to that configuration set entry for groups with special needs.

Figure 7–1 shows three separate directory server instances, each with a different value.

Figure 7–1  Directory Entry Hierarchy Showing Multiple Configuration Set Entries

- An Oracle directory server (cn=osdldap) with:
  - One instance listening on the default port and using `configset0` with SSL disabled
  - A second instance listening on the SSL port and using `configset1` with SSL enabled
- A replication server instance (cn=odsrepld) using `configset0`

---

**Note:** You can run multiple instances if the directory server is on the same computer. For example, you can run one instance in SSL mode and another in non-SSL mode.
Managing Server Configuration Set Entries

Managing Server Configuration Set Entries by Using Oracle Directory Manager

You can use Oracle Directory Manager to view, add, modify, and delete configuration set entries.

Viewing Configuration Set Entries by Using Oracle Directory Manager

To view configuration set entries:

1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance, then Server Management.
3. In the right pane, select an instance, then choose View Properties. A Server Process dialog box appears.

You can see all the parameters for the instance by selecting the tabs across the top of the dialog box. However, you cannot change these parameters in this dialog box. To change them, you must change the configuration set entry on which they are based.

See Also:

- Chapter 17, "Secure Sockets Layer (SSL) and the Directory" for information about configuration parameters for SSL
- Chapter 30, "Oracle Internet Directory Replication Installation and Configuration" for information about configuration parameters for replication
- "OID Configuration Schema Elements" in Oracle Identity Management User Reference for a list and descriptions of the entire set of attributes that are used to configure an instance of a directory server

Note: You cannot change the parameters for an active instance directly. Instead, you must change the parameters in a configuration set entry and save it. After the configuration set entry is saved, use the OID Control Utility restart command to stop current Oracle directory server instances and restart them.

You can change a configuration set entry and start fresh instances that use the new parameters. The changes do not affect the older instances that are still running, however, unless they are restarted.

For information on restarting directory server instances, see "Oracle Internet Directory Server Administration Tools" in Oracle Identity Management User Reference.
Adding Configuration Set Entries by Using Oracle Directory Manager

The first time you add a configuration set entry, you can:

■ Use the default configuration set as a template for the new configuration set entry, then copy from it to make subsequent configuration sets.

■ Add a configuration set entry without copying from an existing one.

Adding a Configuration Set Entry by Copying from the Default Configuration Set Entry

To add configuration set entries by copying the default configuration set entry:

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

2. Select Default Configuration Set.

3. On the toolbar, choose Create Like. The Configuration Sets dialog box displays the General tab page.

4. In the General tab page, fill in the fields. These are described in Table A–40 on page A-24.

5. Select the SSL Settings tab and fill in the fields. These are described in Table A–41 on page A-24.

6. Choose Apply.

7. Restart the server instance for the command to take effect.

Adding a Configuration Set Entry Without Copying from an Existing One

To create a new configuration set entry without copying from a previous configuration set entry:

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

2. Select Default Configuration Set.


4. In the General tab page, fill in the fields. These are described in Table A–40 on page A-24.

5. Select the SSL Settings tab and fill in the fields. These are described in Table A–41 on page A-24.

6. Choose OK.
Modifying Configuration Set Entries by Using Oracle Directory Manager

To modify configuration set entries:

1. In the navigator pane, expand each of the following objects in succession: Oracle Internet Directory Servers, directory server instance, Server Management, Directory Server.
2. Select the configuration set entry you want to modify. The configuration set appears in the group of tab pages in the right pane.
3. In the General tab page, modify the fields. These are described in Table A–40 on page A-24. To save the changes, choose Apply.
4. Select the SSL Settings tab and modify the fields. These are described in Table A–41 on page A-24. To save the changes, choose Apply.
5. Restart the server instance for the command to take effect.

Deleting Configuration Set Entries by Using Oracle Directory Manager

To delete configuration set entries:

1. In the navigator pane, expand each of the following objects in succession: Server Management, Directory Server.
2. Select the configuration set entry you want to delete.
3. On the toolbar, choose Delete.
4. Restart the server instance for the command to take effect.

Managing Server Configuration Set Entries by Using Command-Line Tools

Although changing configuration set entries by using Oracle Directory Manager is desirable, it can sometimes be more convenient to use the available command-line tools—for example, when you want to make the same set of changes across multiple Oracle directory servers.

When you add or modify configuration set entries by using the command-line tools, the input file for adding a new configuration set entry must be written in LDIF. It must contain only the attributes and values that differ from the installed defaults. The directory server uses the attribute values that you establish in the new configuration set entry to override its own existing values for these attributes.

This section contains the following topics:

- Adding Configuration Set Entries by Using ldapadd
Modifying and Deleting Configuration Set Entries by Using ldapmodify

Adding Configuration Set Entries by Using ldapadd

If you are adding a new Oracle directory server instance, then you can either use an existing configuration set entry, or add a new one for the new instance.

To add a new configuration set entry, create an input file, and then load the input file with ldapadd. Follow these steps:

1. Create the input file in a text editor.
   
   Input files must use LDIF format. When you create the input file, you need to define or include only those attributes that differ from the current values in that configuration set entry.

   In this example, the parameter configset2 is the RDN, or local name, of the new entry and the wallet location is: /HOME/test/wallet.

   ```
   dn:cn=configset2, cn=osdldapd, cn=subconfigsubentry
   cn:configset2
   objectclass:orclConfigSet
   objectclass:orclLDAPSubConfig
   objectclass:top
   orclsslauthentication:1
   orclsslenable:1
   orclsslport:5000
   orclsslversion:3
   orclsslwalleturl:file:/HOME/test/wallet
   ```

2. Run ldapadd with an input file.
   
   At the system prompt, type the command to add the input file.

   ```
   ldapadd [options] -f LDIF_file_name
   ```

   See Also:

   - "LDIF File Formatting Rules and Examples" in Oracle Identity Management User Reference
   - The ldapadd command-line tool reference in Oracle Identity Management User Reference for a detailed list of options available with this command
   - "OID Configuration Schema Elements" in Oracle Identity Management User Reference for a description of configuration set entry attributes

Modifying and Deleting Configuration Set Entries by Using ldapmodify

To modify or delete an existing configuration set entry, create an input file containing only the attributes that you want to change, and then load the input file with the ldapmodify command. Follow these steps:

1. Create the input file.
   
   When you create the input file, define or include only those attributes that differ from the installed defaults.

   Input files must have LDIF format.

   In the next example, the parameter 
cn=configset2,cn=osdldapd,cn=subconfigsubentry is the DN, or local
name, of an existing configuration set entry. This example shows how to modify the ORCLSSLPORT parameter to 7000.

dn:cn=configset2, cn=osldapd, cn=subconfigsubentry
changetype: modify
replace: orclsslport
orclsslport: 7000

2. Run ldapmodify referencing the input file.

Type the command to reference the input file at the system prompt.

`ldapmodify [options] -f LDIF_file_name`

See Also:
- "LDIF File Formatting Rules and Examples" in Oracle Identity Management User Reference
- The `ldapmodify` command-line tool reference in Oracle Identity Management User Reference for a more detailed discussion of ldapmodify, and a list of its options
- "OID Configuration Schema Elements" in Oracle Identity Management User Reference for a description of configuration set entry attributes

Setting System Operational Attributes

An operational **attribute**—as opposed to an application attribute—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. You must have super user privileges to set system operational attributes.

This section contains these topics:
- Setting System Operational Attributes by Using Oracle Directory Manager
- Setting System Operational Attributes by Using ldapmodify

See Also:  "Kinds of Attribute Information" on page 3-9

Setting System Operational Attributes by Using Oracle Directory Manager

You can view and set some of the operational attributes for each Oracle directory server to which you are connected by using Oracle Directory Manager. To do this, in the navigator pane, expand Oracle Internet Directory Servers, then select a directory server. System operational attributes appear in the right pane.

Table A–42 on page A-25 describes the fields displayed in Oracle Directory Manager for system operational attributes.

Setting System Operational Attributes by Using ldapmodify

To modify system operational attributes, use ldapmodify. The "Oracle Identity Management LDAP Attribute Reference" in Oracle Identity Management User Reference describes the modifiable system operational attributes.
Managing Naming Contexts

To enable users to search for specific naming contexts, you can publish those naming contexts. This section contains these topics:

- Publishing Naming Contexts by Using Oracle Directory Manager
- Publishing Naming Contexts by Using ldapmodify

To publish a naming context, you specify the topmost entry of each naming context as a value of the `namingContexts` attribute in the root DSE. For example, suppose you have a DIT with three major naming contexts, the topmost entries of which are `c=uk`, `c=us`, and `c=de`. If these entries are specified as values in the `namingContexts` attribute, then a user, by specifying the appropriate filter, can find information about them by searching the root DSE. The user can then focus the search—for example, by concentrating on the `c=de` naming context in particular.

You can use either Oracle Directory Manager or ldapmodify to publish a naming context. The `namingContexts` attribute is multi-valued, so you can specify multiple naming contexts.

To search for published naming contexts, perform a base search on the root DSE with `objectClass=*` specified as a search filter. The retrieved information includes those entries specified in the `namingContexts` attribute.

Before you publish a naming context, be sure that:

- You are a directory administrator with the necessary access to the root DSE.
- The topmost entry of that naming context exists in the directory.

Publishing Naming Contexts by Using Oracle Directory Manager

1. In the navigator pane, expand Oracle Internet Directory Servers and select the directory server on which you want to specify a naming context. The corresponding tab pages for that directory server appear in the right pane.

2. In the System Operational Attributes tab page, in the Naming Contexts field, enter the topmost DN of the naming context you want to publish. You can also choose Browse to open a search window.

3. Choose Apply.

Publishing Naming Contexts by Using ldapmodify

The following sample LDIF file specifies the entry `c=uk` as a naming context.

```
 dn: 
 changetype: modify 
 add: namingcontexts 
 namingcontexts: c=uk 
```

Managing Super Users, Guest Users, and Proxy Users

This section contains these topics:

- About Super Users, Guest, Users, and Proxy Users
Managing Super Users, Guest Users, and Proxy Users by Using Oracle Directory Manager

Managing Super Users, Guest Users, and Proxy Users by Using ldapmodify

About Super Users, Guest, Users, and Proxy Users

Table 7-1 on page 7-9 defines super users, guest users, and proxy users.

You can use either Oracle Directory Manager or ldapmodify to administer the user names and passwords for each of these users.

Table 7-1 Definitions of Super User, Guest User, and Proxy User

<table>
<thead>
<tr>
<th>Type of User</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Super user</td>
<td>A special directory administrator with full access to directory information. The default user name of the super user is orcladmin; the default password is welcome.</td>
</tr>
<tr>
<td>Guest user</td>
<td>One who is not an anonymous user, and, at the same time, does not have a specific user entry. The default user name for a guest user is guest; the default password is guest.</td>
</tr>
<tr>
<td>Proxy user</td>
<td>A user typically in an environment with a middle tier such as a firewall, an application such as Oracle Delegated Administration Services, or a RADIUS server. The default user name for a proxy user is proxy; the default password is proxy.</td>
</tr>
</tbody>
</table>

See Also: "Indirect Authentication" on page 16-5 for more information about proxy users

See Also: Chapter 18, "Directory Access Control" for information on how to set access rights

Note: You can log on to the Oracle Directory Manager without giving a user name or password. If you do this, you have the privileges specified for an anonymous user. Anonymous users should have very limited privileges.

Managing Super Users, Guest Users, and Proxy Users by Using Oracle Directory Manager

Note: The passwords for super users, guest users, and proxy users are encrypted by default. You cannot modify them to send them in the clear.

To set a user name or password for a super user, a guest user, or a proxy user by using Oracle Directory Manager:

1. In the navigator pane, expand Oracle Internet Directory Servers, then select a directory server. The group of tab pages for that server appear in the right pane.

2. Select the System Passwords tab. This page displays the current user names and passwords for each type of user. Note that passwords are not displayed in the password fields.

3. Edit the appropriate field in the System Passwords tab page as described in Table A–43 on page A-29. To save your changes, choose Apply.
Managing Anonymous Binds

Managing Super Users, Guest Users, and Proxy Users by Using ldapmodify

To set or modify a user name or password for a super user, a guest user, or a proxy user, use ldapmodify to modify the appropriate attribute.

To set or modify a user name or password for a super user, a guest user, or a proxy user, use ldapmodify to modify the appropriate attribute.

Table 7–2 Names, Passwords, and Attributes for Super, Guest, and Proxy Users

<table>
<thead>
<tr>
<th>User Name</th>
<th>Password</th>
<th>Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>Super user name</td>
<td>orclsupassword</td>
<td>orclsuname</td>
</tr>
<tr>
<td>Guest user name</td>
<td>orclgupassword</td>
<td>orclguname</td>
</tr>
<tr>
<td>Proxy user name</td>
<td>orclprpassword</td>
<td>orclprname</td>
</tr>
</tbody>
</table>

For example, to change the password of the super user to superuserpassword, use ldapmodify to modify the DSE by using an LDIF file containing the following:

dn:
changetype: modify
replace: orclsupassword
orclsupassword: superuserpassword

See Also: The ldapmodify command-line tool reference in Oracle Identity Management User Reference for ldapmodify syntax and usage notes.

Managing Anonymous Binds

The Oracle Directory Server can be configured to allow or disallow anonymous bind. This behavior is controlled by the orclAnonymousBindsFlag attribute of the root DSE entry. Table 7–3 lists the allowed values for orclAnonymousBindsFlag and the resulting directory server behavior.

Table 7–3 OrclAnonymousBindsFlag Value and Directory Server Behavior

<table>
<thead>
<tr>
<th>orclAnonymousBindsFlag Value</th>
<th>Directory Server Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Disallows anonymous bind</td>
</tr>
<tr>
<td>1</td>
<td>Allows anonymous bind (default)</td>
</tr>
<tr>
<td>2</td>
<td>Allows anonymous bind but allows only search operations on root DSE entry for anonymous users</td>
</tr>
</tbody>
</table>

By default, the directory server allows anonymous bind. That is, the orclAnonymousBindsFlag attribute is set to 1.

The following example shows how to disable anonymous bind using the ldapmodify command-line tool:

```
ldapmodify -h hostname -p port -D cn=orcladmin -w super_user_pwd <<EOF
dn:
changetype: modify
replace: orclAnonymousBindsFlag
orclAnonymousBindsFlag: 0
EOF
```
Changing the Password to the Oracle Internet Directory Database Server

See Also:

- "Setting System Operational Attributes" on page 7-7
- Table A–42, "System Operation Attributes Displayed in Oracle Directory Manager" on page A-25

Viewing Active Server Instance Information

To view information about any active directory server instance—including type, instance number, debug level, host name, and configuration parameters—use Oracle Directory Manager. To do this:

1. In the navigator pane, expand Oracle Internet Directory Servers and select a directory server. The group of tab pages for that directory server instance appear in the right pane.

2. Select the Server Management tab. This displays basic information—namely, type, instance number, debug level, and host name—for all active directory server instances.

3. To see configuration parameters for a particular directory server instance, select the directory server instance, then choose View Properties. The Server Process dialog box displays configuration parameters for the directory server instance you selected. Note that you cannot change configuration parameters in this dialog box. To change them, you must change the configuration set entry on which they are based.

See Also: "Managing Server Configuration Set Entries by Using Oracle Directory Manager" on page 7-3 for instructions on changing configuration set entries

Closing Idle LDAP Connections

You can specify the number of minutes that LDAP connections remain idle before closing. To do this, you set a value for the orclLDAPconnTimeout attribute described in "Oracle Identity Management LDAP Attribute Reference" in Oracle Identity Management User Reference.

As of 10g (10.1.4.0.1), you can only set this for users who are not configured for operation statistics tracking as described in "Capabilities of Oracle Internet Directory Server Manageability" on page 14-14.

Note: The Oracle Internet Directory server processes idle connection timeout at an interval equal to the configured connection timeout value. This minimizes performance overhead. As a result, sometimes dropping of a connection may take longer than the pre configured connection timeout value. It is guaranteed to take no more time than twice this value.

Changing the Password to the Oracle Internet Directory Database Server

The Oracle Internet Directory uses a password when connecting to its own designated Oracle database. The default for this password when you install Oracle Internet Directory is the same as that for the Oracle Application Server administrator. You can change this password by using the OID Database Password Utility.
Dereferencing Alias Entries

Because entries sometimes have distinguished names that are fairly long and cumbersome, Oracle Internet Directory makes it easier to administer them by using alias objects. When someone looks up—that is, references—an object by using an alias, the alias is dereferenced, and what is returned is the object to which the alias points. For example, the alias, Server1, can be dereferenced so that it points to the fully qualified DN—namely, dc=server1,dc=us,dc=myCompany,dc=com. This feature also enables you to devise structures that are not strictly hierarchical.

This section provides examples of how to add, search for, and modify alias entries, and it includes a list of messages. It contains these topics:

- About Alias Entries
- Examples: Using Alias Entry Dereferencing
- Success and Error Messages

About Alias Entries

An alias entry uses the object class alias to distinguish it from object entries in a directory. The definition of that object class is as follows:

(2.5.6.1 NAME 'alias' SUP top STRUCTURAL MUST aliasedObjectName)

An alias entry also contains the aliasedObjectName attribute that, in turn, contains the DN of the object to which it is pointing. The definition of that attribute is as follows:

(2.4.5.1 NAME 'aliasedObjectName' EQUALITY distinguishedNameMatch SYNTAX 1.3.6.1.4.1.1466.115.121.1.12 SINGLE-VALUE)

Figure 7–2 and the accompanying text provides an example of alias entry dereferencing.

Figure 7–2 Alias Entries Example

In Figure 7–2, ou=uk sales,ou=global sales,o=myCompany,c=us is an alias entry pointing to the ou=sales,o=myCompany,c=uk entry.
When anyone references `ou=uk sales,ou=global sales,o=oracle,c=us`, the directory server automatically reroutes them to the real entry, `ou=sales,o=oracle,c=uk`.

**Examples: Using Alias Entry Dereferencing**

This section contains these examples:

- **Example: Adding an Alias Entry**
- **Examples: Searching the Directory with Alias Entries**
- **Example: Modifying Alias Entries**

**Example: Adding an Alias Entry**

To add an alias entry, you create a normal entry in LDIF and an alias entry pointing to the real entry. Following the steps in this example produces the tree in Figure 7–3 on page 7-14.

1. Create a sample LDIF file, `My_file.ldif`, with the following entries:

   ```ldif
   dn: c=us
   c: us
   objectclass: country
   
   dn: o=MyCompany, c=us
   o: MyCompany
   objectclass: organization
   
   dn: ou=Area1, c=us
   objectclass: alias
   aliasedObjectName: o=MyCompany, c=us
   
   dn: cn=John Doe, o=MyCompany, c=us
   cn: John Doe
   sn: Doe
   objectclass: person
   
   dn: cn=President, o=MyCompany, c=us
   objectclass: alias
   aliasedObjectName: cn=John Doe, o=MyCompany, c=us
   ```

2. Add these entries to the directory by using the following command:

   ```bash
   ldapadd -p port -h host -f My_file.ldif
   ```

**Note:** When you add an alias entry whose parent is an alias entry, the directory server returns an error.

**See Also:** Entry Alias Dereferencing Messages on page 7-16 for error messages.
Dereferencing Alias Entries

In Figure 7–3, the letter A represents an alias entry, where:

- `ou=Area1` is an alias pointing to `o=MyCompany`
- `cn=President` is an alias pointing to `cn=John Doe`

Examples: Searching the Directory with Alias Entries

In each search you specify, there are flags you can set. The search is performed based on the flag you specify.

The flags pertaining to alias dereferencing are `-a never` and `-a find`.

By default, the dereference flag in `ldapsearch` is `-a never` and thus the directory server does not perform any dereferencing for alias entries.

Note: As of 10g (10.1.4.0.1), Oracle Internet Directory does not support the following options:

- The `ldapsearch` options `-a search` and `-a always`.
- The `extensibleMatch` feature described in RFC 2251.

Example: Searching the Base  A base search finds the top level of the alias entry you specify.

This example shows a base search of `ou=Area1,c=us` with a filter of "objectclass=*" with the dereferencing flag set to `-a find`.

```
ldapsearch -p port -h host -b "ou=Area1,c=us" -a find -s base "objectclass=*"
```

The directory server, during the base search, looks up the base specified in the search request and returns it to the user. However, if the base is an alias entry and, as in the example, `-a find` is specified in the search request, then the directory server automatically dereferences the alias entry and returns the entry it points to. In this example, the search dereferences `ou=Area1,c=us`, which is an alias entry, and returns `o=MyCompany,c=us`.

Example: Searching One-Level  A one-level search finds only the child to the base level you specify.

This example shows a one-level search of "ou=Area1,c=us" with a filter of "objectclass=*" with the dereferencing flag set to `-a find`.

```
ldapsearch -p port -h host -b "ou=Area1,c=us" -a find -s one "objectclass=*"
```

The directory server performs the search in two steps:

1. It searches for the base that is specified in the search request.
When it locates the base, it looks up all one-level entries under this base and returns entries that match the filter criteria.

In the example, `-a find` is specified in the search request, and thus the directory server automatically dereferences while looking up the base (the first step), but does not dereference alias entries that are one level under the base. Therefore, the search dereferences `ou=Area1,c=us`, which is an alias entry, and then looks up one-level entries under `o=MyCompany,c=us`. One of the one-level entries is `cn=President,o=MyCompany,c=us` that is not dereferenced and is returned as is.

Thus, the search returns `cn=President,o=MyCompany,c=us` and `cn=John Doe,o=MyCompany,c=us`.

**Example: Searching a Subtree**  A subtree search finds the base, children, and grand children.

This example shows a subtree search of "ou=Area1,c=us" with a filter of "objectclass=*" with the dereferencing flag set to `-a find`.

```
ldapsearch -p port -h host -b "ou=Area1,c=us" -a find -s one "objectclass=*"
```

The directory server performs the search in two steps.

1. It searches for the base that is specified in the search request.
2. When it locates the base, then it looks up all entries under this base and returns entries that match the filter criteria.

In the example, `-a find` is specified in the search request, and thus the directory server automatically dereferences while looking up the base (the first step), but does not dereference alias entries that are under the base. Therefore, the search dereferences `ou=Area1,c=us`, which is an alias entry, and then looks up entries under `o=MyCompany,c=us`. One of the entries is `cn=President,o=MyCompany,c=us` that is not dereferenced and is returned as is.

Thus, the search returns:

- `o=MyCompany,c=us`
- `cn=John Doe,o=MyCompany,c=us`
- `cn=President,o=MyCompany,c=us`

**Example: Modifying Alias Entries**

This example shows how to modify alias entries. It creates a sample LDIF file, `My_file.ldif` with following entries:

```
dn: cn=President, o=MyCompany, c=us
changetype : modify
replace: aliasedobjectname
aliasedobjectname: cn=XYZ, o=MyCompany, c=us
```

Modify the alias entry using the following command:

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

**Success and Error Messages**

Table 7–4 lists the messages related to alias entry dereferencing and the corresponding meaning for each message.
Locating Directory Servers in a Distributed Environment

To perform an operation on a particular entry, a client must be able to find the server in which that entry resides. In a distributed environment, information about the location of a server can be available in one of two ways:

- Statically, in the directory server usage file (`ldap.ora`) stored on the client host.
- Dynamically, by using the domain name system (DNS). In this case, the information about server location is stored and managed in a central domain name server. The client, at request processing time, retrieves this information from the domain name server dynamically.

This section discusses these two methods of locating server information. It contains these topics:

- **Static Directory Server Discovery by Using the Directory Server Usage File (`ldap.ora`)**
- **Dynamic Directory Server Discovery by Using the Domain Name System (DNS)**

**See Also:**

- "A DNS RR for specifying the location of services (DNS SRV)", Internet RFC 2782 at [http://www.ietf.org](http://www.ietf.org)

### Table 7–4 Entry Alias Dereferencing Messages

<table>
<thead>
<tr>
<th>Message</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alias Problem</td>
<td>Either of the following have occurred:</td>
</tr>
<tr>
<td></td>
<td>- An alias was dereferenced, but it did not point to an entry in the DIT.</td>
</tr>
<tr>
<td></td>
<td>- The user tries to add an alias entry whose parent is an alias.</td>
</tr>
<tr>
<td>Alias Dereferencing Problem</td>
<td>The user cannot dereference an alias because of access control issues.</td>
</tr>
<tr>
<td>No Such Object</td>
<td>The server cannot find the base DN specified in the search request.</td>
</tr>
<tr>
<td>Invalid DN Syntax</td>
<td>When adding or modifying an alias entry, if the value specified for <code>aliasedObjectName</code> has invalid DN syntax, then the directory server returns this error message to the client.</td>
</tr>
<tr>
<td>Success</td>
<td>The client operation successfully completes.</td>
</tr>
<tr>
<td></td>
<td>When the dereferenced target does exist but does not match the filter specified in the search request, the server returns a success message with no matched entry.</td>
</tr>
<tr>
<td>Insufficient Access Rights</td>
<td>The user does not have access to the dereferenced entry.</td>
</tr>
</tbody>
</table>

**Static Directory Server Discovery by Using the Directory Server Usage File (`ldap.ora`)**

Using this method, when a client seeks to perform an operation on a directory entry, it obtains directory server location information from the directory server usage file (`ldap.ora`) stored on the client host. This file contains configuration parameters that specify:

- The type of directory server—for example, Oracle Internet Directory, Microsoft Active Directory, SunONE Directory Server, and so forth.
- The location of the directory server
- The default directory entry that the client or server will use to look up or configure connect identifiers for connections to database services

The file ldap.ora resides in the file system of the LDAP client. The client looks for this file in the following file system directories, in order of precedence:

1. The directory pointed to by the LDAP_ADMIN environment variable
2. The directory ORACLE_HOME/ldap/admin (or, on Microsoft Windows, ORACLE_HOME\ldap\admin)
3. The directory pointed to by the TNS_ADMIN environment variable
4. The directory ORACLE_HOME/network/admin (or, on Microsoft Windows, ORACLE_HOME\network\admin)

If the file ldap.ora is present in more than one location, then the location having higher precedence is honored.

Using the static method to discover a directory server can increase management overhead. For example, because the ldap.ora file is stored on the client host, the administrator must update that file on every client whenever the host name or port number of a directory server is changed. To avoid this increased overhead, you can enable an application to discover directory servers dynamically by using the domain name system (DNS).

### Dynamic Directory Server Discovery by Using the Domain Name System (DNS)

The domain name system (DNS) is a dynamic way of locating domain names and translating them into the actual addresses of computers. This translation process is handled by a central domain name server, which contains information about the locations of directory servers.

Once a network administrator has entered the necessary information about directory server locations in a domain name server, clients can retrieve that information from that server instead of from ldap.ora files.

For a client to locate a directory server by using DNS, the following steps must have been completed:

- The network administrator must have entered a DNS Service Location Record (SRV) into the domain name server.
- The client application must have been enabled to map distinguished names to domain names.

### How a Client Locates a Directory Server by Using DNS

To find the directory server on which an entry resides, a client communicates with the domain name server. Specifically, it provides to the domain name server a domain name. The domain name specifies where the needed directory server is located.

To generate the domain name, the client extracts the domain component from the DN entered by the user. For example, in the DN cn=John Doe,ou=accounting,dc=example,dc=net, the domain component is dc=example,dc=net. That domain component represents the server on which the requested entry resides. The client then converts that domain name component to a domain name in a format recognized by the domain name server, namely, example.net.
Figure 7–4 and the accompanying text show the process of locating a directory server from the perspective of a client.

**Figure 7–4  A Client Locating a Directory Server by Using DNS**

1. A user wanting to perform an operation on a directory entry, enters into the client the distinguished name (DN) of that entry—for example, `cn=John Doe,ou=accounting,dc=example,dc=net`.

2. To communicate with the domain name server, the client converts the domain component of the DN to a domain name. In the example used here, the client would convert the domain component of that DN—namely, `dc=example,dc=net`—to the domain name `example.net`.

3. The client queries the domain name server for SRV resource records having the specified domain name.

4. The domain name server returns the SRV resource records that match the specified domain name. These resource records contain the host name information of the directory server containing the requested entry. If the domain name server is not able to find any matching SRV resource records, then it returns an error message.

5. The client parses the records. It extracts the directory host name information from these records and returns it to the user.

**See Also:**

**Note:** The domain name server either stores all the necessary SRV records locally, or obtains them from other domain name servers. If the domain name server cannot find the requested information, then it returns an error message. It does not return a referral to another domain name server.

---

**Registering a Directory Server with the Domain Name System**

Registering server location information for a directory server involves entering a DNS service location record (SRV) into the domain name server. The SRV record contains:

- The DNS name of the server that provides the LDAP service
- The corresponding port number
- Parameters that enable the client to choose an appropriate server from multiple servers
The SRV resource record enables administrators to use several servers for a single domain, to move services from host to host easily, and to designate some hosts as primary servers for a service and others as backups.

The format of the SRV record can be either specific to Oracle Internet Directory servers or standard. For information about Oracle Internet Directory servers, the Oracle Internet Directory-specific format is preferred. When a client first queries a domain name server, it looks for SRV records that have the Oracle Internet Directory-specific format. If it does not find any with this format, then it queries for SRV records that have the standard format.

**The Oracle Internet Directory-Specific Format for SRV Records**

The Oracle Internet Directory-specific format is:

```
_Service._Proto._Product._Domain TTL Class Type Priority Weight Port Target
```

Table 7–5 describes the arguments. The following is an example of an SRV record that uses the Oracle Internet Directory-specific format.

```
_ldap._tcp._oid.acme.com 0 IN SRV 0 1 389 ldap.acme.com
```

**The Standard Format for SRV Records**

The standard format is:

```
_Service._Proto._Domain TTL Class Type Priority Weight Port Target
```

Table 7–5 describes the arguments. The following is an example of an SRV record for a non-SSL-based directory server that uses the standard format.

```
_ldap._tcp.acme.com 0 IN SRV 0 1 389 ldap.acme.com
```

**Table 7–5 Arguments in a Service Location Record (SRV)**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service</td>
<td>For a non-SSL-based server, the value for this argument is ldap. For an SSL-based server, the value is ldaps.</td>
</tr>
<tr>
<td>Proto</td>
<td>The value is always tcp.</td>
</tr>
<tr>
<td>Product</td>
<td>The value is always oid.</td>
</tr>
<tr>
<td>Domain</td>
<td>The domain name. It is usually obtained by converting the DN of the naming context mastered by the directory server into a domain name.</td>
</tr>
<tr>
<td>TTL</td>
<td>Time to live. This argument has the standard DNS meaning. It specifies how long the resource record may be cached before the source of the information is again consulted.</td>
</tr>
<tr>
<td>Class</td>
<td>This argument has the standard DNS meaning. SRV records occur in the IN class.</td>
</tr>
<tr>
<td>Type</td>
<td>For all SRV records, the value for this argument is SRV.</td>
</tr>
<tr>
<td>Priority</td>
<td>The priority of the directory server. A client must attempt to contact the target host with the lowest-numbered priority.</td>
</tr>
</tbody>
</table>
Locating Directory Servers in a Distributed Environment

Table 7–5 (Cont.) Arguments in a Service Location Record (SRV)

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
</table>
| Weight   | A server selection mechanism, this argument specifies a relative weight for entries with the same priority. If multiple SRVs have the same priority, then they are ordered according to the following protocol:  
1. To select a target to be contacted next, arrange in any order all SRV resource records that have not yet been ordered—but place all those with weight 0 at the beginning of the list.  
2. Compute the sum of the weights of those resource records, and with each resource record associate the running sum in the selected order.  
3. Choose a uniform random number between 0 and the sum computed (inclusive), and select the resource record whose running sum value is the first in the selected order that is greater than or equal to the random number selected. The target host specified in the selected SRV resource record is the next one to be contacted by the client.  
4. Remove this SRV resource record from the set of the unordered SRV resource records.  
5. Apply the described algorithm to the unordered SRV resource records to select the next target host.  
6. Continue the ordering process until there are no unordered SRV resource records.  
7. Repeat this process for each priority. |
| Port     | The port on target host for the directory service. |
| Target   | The domain name of the host on which the directory server is running. |

**Note:** If the directory server is moved to a different host or is run on different port, then the corresponding SRV resource record must be updated accordingly.
This chapter explains how to view, add, modify, and delete entries. It contains these topics:

- Managing Entries by Using Oracle Directory Manager
- Managing Entries by Using Command-Line Tools
- Managing Knowledge References and Referrals

See Also: Chapter 3, "Directory Concepts and Architecture" for an overview of directory entries, directory information trees, distinguished names, and relative distinguished names

Bulk tools are discussed in Chapter 9, "Using Bulk Tools".

Managing Entries by Using Oracle Directory Manager

This section contains these topics:

- Searching for Entries by Using Oracle Directory Manager
- Viewing Attributes for a Specific Entry by Using Oracle Directory Manager
- Adding Entries by Using Oracle Directory Manager
- Modifying Entries by Using Oracle Directory Manager
- Managing Entries with Attribute Options by Using Oracle Directory Manager

Searching for Entries by Using Oracle Directory Manager

You can display all entries by using the navigator pane, or search for one or more specific entries by using the Oracle Directory Manager search feature.

To display an entry, in the navigator pane, expand Oracle Internet Directory Servers, then directory server instance, then Entry Management.

The root of the tree is listed first, then the second level, and so forth, moving from left to right. The subtree lists the RDN of each entry in hierarchical order. To see the lower level entries within any subtree, click the plus sign (+) to the left of the parent entry.

To search for a directory entry:

1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance, then Entry Management. The Search fields appear in the right pane.

2. In the Root of the Search field, enter the DN of the root of your search.
For example, suppose you want to search for an employee who works in the Manufacturing division in the IMC organization in the Americas. The DN of the root of your search would be:

ou=Manufacturing,ou=Americas,o=IMC,c=US

You would therefore type that DN in the Root of the Search text box.

You can also select the root of your search by browsing the directory information tree (DIT). To do this:

a. Click Browse to the right of the Root of the Search field. The Select Distinguished Name (DN) Path: Tree View dialog box appears.
b. Click the plus sign (+) next to tree view to display its entries.
c. Continue navigating to the entry that represents the level you want for the root of your search.
d. Select that entry, then click OK. The DN for the root of your search appears in the Root of the Search text box in the right pane.

3. In the Max Results (entries) box, type the maximum number of entries you want your search to retrieve. The default is 200. The directory server retrieves the value you set, up to 1000.

4. In the Max Search Time (seconds) box, type the maximum number of seconds for the duration of your search. The value you enter here must be at least that of the default, namely, 25. The directory server searches for the amount of time you specify, up to one hour.

5. In the Search Depth list, select the level in the DIT to which you want to search. The options are:
   - **Base**: Retrieves a particular directory entry. Along with this search depth, you use the search criteria bar to select the attribute objectClass and the filter Present.
   - **One Level**: Limits your search to all entries beginning one level down from the root of your search.
   - **Subtree**: Searches entries within the entire subtree, including the root of your search.

6. In the Search Criteria box, use the lists and text fields on the search criteria bar to focus your search.
   a. From the list at the left end of the search criteria bar, select an attribute of the entry for which you want to search. Because not all attributes are used in every entry, be sure that the attribute you specify actually corresponds to one in the entry for which you are looking. Otherwise, the search will fail.
   b. From the list in the middle of the search criteria bar, select a filter. Options described in Table A–45 on page A-31.
   c. In the text box at the right end of the search criteria bar, type the value for the attribute you just selected. For example, if the attribute you selected was cn, you could type the particular common name you want to find.

7. To further refine your search, use the buttons in the Search Criteria box to enhance the search criteria bar. These are described Table A–46 on page A-31.

8. Click Search. The results of your search appear in the Distinguished Name box.
Managing Entries by Using Oracle Directory Manager

Viewing Attributes for a Specific Entry by Using Oracle Directory Manager

Once you have displayed the results of your search, click the entry whose attributes you want to view. An Entry dialog box displays the attributes for that entry.

Some attributes can also be DNs. For example, one attribute for a given employee might be that employee's manager who, in turn, has a DN. In this case, when you display the Entry dialog box for the employee, you would see a Browse button next to the Manager text box. To find information about that manager, click Browse to display the Directory: Entry Management dialog box, then follow the steps mentioned in "Searching for Entries by Using Oracle Directory Manager" on page 8-1.

Adding Entries by Using Oracle Directory Manager

This section tells you how to add entries for users and groups.

---

**Note:** When you add or modify an entry, the Oracle directory server does not verify the syntax of the attribute values in the entry.

---

Adding a New Entry by Using Oracle Directory Manager

To add or delete entries with Oracle Directory Manager, you must have write access to the parent entry and you must know the DN for the new entry.

To add a new entry:

1. In the navigator pane, expand each of the following objects in succession: Oracle Internet Directory Servers, directory server instance.
2. Select Entry Management.
3. On the toolbar, click Create. The New Entry dialog box appears.
4. In the Distinguished Name field, type the full DN. You can also click Browse to locate and select the DN of the parent for the entry you want to add. The entry you select appears in the Distinguished Name field. To the left of that parent DN, type the RDN for your new entry, followed by a comma.
5. To specify an object class for the new entry, next to the Object Classes box, click Add. The Super Class Selector dialog box appears.

**Note:** You must assign user entries to the inetOrgPerson object class in order for the entries to appear in the Oracle Internet Directory Self-Service Console.

6. In the Super Class Selector dialog box, select an object class, then click Select. As you select from the object class list, mandatory and optional attributes populate the windows in the tab pages in the lower half of the New Entry dialog box. You
must enter values into the mandatory attributes fields. You are not required to enter values into the optional attributes fields.

7. When you have selected the object classes and provided values for the appropriate attributes, click OK.

**Adding an Entry by Copying an Existing Entry in Oracle Directory Manager**

You can use Oracle Directory Manager to create a new entry by copying from an existing entry and changing its DN. When you do this, you should also change the attributes, such as name and address, so that they correspond to the new DN. To add an entry, you must have write access to its parent.

**Tip:** You can find a template for the new DN by looking up other similar entries in the search pane.

To add an entry by copying an existing entry:

1. In the navigator pane, expand each of the following objects in succession: Oracle Internet Directory Servers, directory server instance.
2. Select Entry Management.
3. In the right pane, the search interface appears. Use it to search for an entry that you want to use as a template.
4. From the entries retrieved, double-click one that you want to use as your template. The Entry dialog box for that entry appears.
5. In the Entry dialog box, click Create Like. A New Entry: Create Like dialog box appears.
6. Change critical fields to tailor this entry to the one that you want to create. You must always change the DN and the common name in this operation, or the pane will not save your new entry data. For example, if you create an entry for Henri Latrobe by using the entry for Henri Latour as the template, then you have to change `cn=Henri Latour in the DN to `cn=Henri Latrobe`. You also must change any other attributes that must be unique, such as employee number and telephone number.
7. Click OK to save your changes.

**See Also:** The online help for this dialog box for details about adding information into fields

**Example: Adding a User Entry by Using Oracle Directory Manager**

In this example, we create a user named Anne Smith and assign her a password.

1. Login as the administrator.
2. In the navigator pane, expand each of the following objects in succession: Oracle Internet Directory Servers, directory server instance.
3. Select Entry Management.
4. On the toolbar, click Create. The New Entry dialog box appears.
5. In the Distinguished Name field, type the full DN. You can also click the Browse button to locate the DN of the parent for this entry, then type the RDN—namely, `cn=Anne Smith`—followed by a comma, to the left of that parent DN.
6. To the right of the Object Classes box, click **Add**. The Super Class Selector dialog box appears.

7. In the Super Class Selector dialog box, select the **person** object class, then click **Select**. This returns you to the New Entry dialog box.

8. In the New Entry dialog box, click the **Optional Properties** tab, and scroll to the User Password window.

9. Type the password for Anne Smith.

### See Also:

- "Searching for Entries by Using Oracle Directory Manager" on page 8-1
- "Managing Group Entries" on page 13-6
- "Security Groups" on page 18-3
- "Globalization Support" on page 3-15 and Chapter 18, "Directory Access Control" for information about access privileges

### Modifying Entries by Using Oracle Directory Manager

You can add auxiliary object classes to an existing entry.

You can add optional, but not mandatory, attributes to an object class already in use by entries. If you add optional attributes to an object class already in use, then no special rules apply, and they are added as empty attributes to those entries.

**Note:** When you add or modify an entry, the Oracle directory server does not verify the syntax of the attribute values in the entry.

To modify an entry:

1. Perform a search for the entry you want to modify as described in "Searching for Entries by Using Oracle Directory Manager" on page 8-1.

2. In the **Distinguished Name** box of the right pane, select the entry you want to modify.

3. Click **Edit**. The Entry dialog box appears.

4. Modify the appropriate fields, then choose **Select the Properties** tab page. If you do not see the attributes you want to add or modify, then, at the top of the tab page, select **View Properties: All**.

5. In the **Properties** tab page, modify the values of any editable attributes.

6. Click **Apply**.

### Example: Modifying a User Entry by Using Oracle Directory Manager

In this example, we modify the password for the entry we created for Anne Smith in the section "Example: Adding a User Entry by Using Oracle Directory Manager" on page 8-4.

1. Perform a search for the Anne Smith entry.
2. In the right pane, in the **Distinguished Name** box, select the entry for Anne Smith.

3. Click **Edit**.

4. In the Entry dialog box, scroll to the User Password window and modify the value.

5. Click **OK**.

**Managing Entries with Attribute Options by Using Oracle Directory Manager**

This section tells you how to add, modify, and delete attribute options.

*See Also:* "Searching for Entries by Using Oracle Directory Manager" on page 8-1 for instructions on searching for entries with attribute options

**Adding an Attribute Option to an Existing Entry by Using Oracle Directory Manager**

*Note:* In Oracle Internet Directory 10g (10.1.4.0.1), Oracle Directory Manager does not allow you to add an attribute option to an entry when you create the entry. You can use Oracle Directory Manager to add attribute options only to already existing entries.

To add an attribute option to an existing entry:

1. In the navigator pane, expand each of the following objects in succession: **Oracle Internet Directory Servers**, **directory server instance**, and **Entry Management**.

2. Select the entry to which you want to add an attribute option. The corresponding tab pages appear in the right pane.

3. In the right pane, in the **Properties** tab page, in the **View Properties** field, select **Advanced**. The **Properties** tab page changes accordingly.

4. In the **Attribute** field, select the attribute to which you want to add the option, for example, **ou**.

5. In the **Attribute Options** field, enter the attribute option, for example, **lang-en**.

6. In the **Attribute Value** field, enter the value of the attribute option you just specified, for example, **Server Technologies**. To add more than one attribute value for the specified attribute option, separate the values by using a semicolon.

7. Click **Apply**.

**Modifying an Attribute Option by Using Oracle Directory Manager**

To modify an attribute option:

1. In the navigator pane, expand each of the following objects in succession: **Oracle Internet Directory Servers**, **directory server instance**, and **Entry Management**.

2. Select the entry whose attribute option you want to modify. The corresponding tab pages appear in the right pane.

3. In the **Properties** tab page, in the **View Properties** field, select either **Only Non-null Values** or **All**.

4. Scroll to the field containing the attribute option you want to modify.

5. Modify the value in the field.
6. Click Apply.

Deleting an Attribute Option by Using Oracle Directory Manager

To delete an attribute option:

1. In the navigator pane, expand each of the following objects in succession: Oracle Internet Directory Servers, directory server instance, and Entry Management.

2. Select the entry from which you want to delete an attribute option. The corresponding tab pages appear in the right pane.

3. In the Properties tab page, in the View Properties field, select either Only Non-null Values or All.

4. Scroll to the field containing the attribute option you want to delete.

5. Delete the value in the field.

6. Click Apply.

Managing Entries by Using Command-Line Tools

This section points you to the command-line tools you can use in managing entries. It also provides several examples of entry management by using command-line tools. It contains these topics:

- Command-Line Tools for Managing Entries
- Managing Entries with Attribute Options by Using Command-Line Tools

Bulk tools are discussed in Chapter 9, "Using Bulk Tools".

Command-Line Tools for Managing Entries

Table 8–1 lists each of the command-line tools for managing entries, and tells you where to find syntax and usage notes for each one.

<table>
<thead>
<tr>
<th>Tool</th>
<th>Task(s)</th>
<th>Syntax and Usage Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ldapadd</td>
<td>Add entries one at a time.</td>
<td>The ldapadd command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td></td>
<td>Add new configuration set entries.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Configure a server with an input file.</td>
<td></td>
</tr>
<tr>
<td>ldapaddmt</td>
<td>Add several entries concurrently by using this shared server tool.</td>
<td>The ldapaddmt command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>ldapbind</td>
<td>Authenticate a user or client to a directory server.</td>
<td>The ldapbind command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td></td>
<td>Verify that you can connect a client to a server.</td>
<td></td>
</tr>
<tr>
<td>ldapcompare</td>
<td>Compare attribute values you specify with those in a directory entry.</td>
<td>The ldapcompare command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>ldapdelete</td>
<td>Delete entries.</td>
<td>The ldapdelete command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
</tbody>
</table>
Example: Adding a User Entry by Using ldapadd
The following example shows an LDIF file, named entry.ldif, for the entry for an employee named John:

```
dn: cn=john, c=us
objectclass: top
objectclass: person
objectclass: organizationalPerson
objectclass: inetOrgPerson
cn: john
cn;lang-fr:Jean
cn;lang-en-us:John
sn: Doe
jpegPhoto: /photo/john.jpg
userpassword: welcome
```

This file contains the cn, sn, jpegPhoto, and userpassword attributes.

For the cn attribute, it specifies two options: cn;lang-fr, and cn;lang-en-us. These options return the common name in either French or American English.

For the jpegPhoto attribute, it specifies the path and file name of the corresponding JPEG image you want to include as an entry attribute.

Note:
- When you add or modify an entry, the Oracle directory server does not verify the syntax of the attribute values in the entry.
- You cannot insert a tilde (~) in a user name.

Example: Modifying a User Entry by Using ldapmodify
The following example changes the password for a user named Audrey from welcome to audreyspassword. As in the previous example, the data for this user entry is in the entry.ldif file. This file contains the following:

```
dn: cn=audrey,c=us
changetype: modify
replace: userpassword
userpassword: audreyspassword
```

Table 8–1 (Cont.) Command-Line Tools for Managing Entries

<table>
<thead>
<tr>
<th>Tool</th>
<th>Task(s)</th>
<th>Syntax and Usage Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ldapmoddn</td>
<td>Modify the DN or RDN of an entry. Rename an entry or a subtree. Move an entry or a subtree under a new parent.</td>
<td>The ldapmoddn command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>ldapmodify</td>
<td>Create, update, and delete attribute data for an entry. Modify configuration set entries. Modify DN or RDN of an entry.</td>
<td>The ldapmodify command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>ldapmodifymt</td>
<td>Modify several entries concurrently by using this shared server tool.</td>
<td>The ldapmodifymt command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
<tr>
<td>ldapsearch</td>
<td>Search for directory entries.</td>
<td>The ldapsearch command-line tool reference in Oracle Identity Management User Reference</td>
</tr>
</tbody>
</table>
Issue this command to modify the file:

```
ldapmodify -p 389 -v -f entry.ldif
```

where -v specifies verbose mode.

---

**Note:** When you add or modify an entry, the Oracle directory server does not verify the syntax of the attribute values in the entry.

---

Managing Entries with Attribute Options by Using Command-Line Tools

This section provides examples of how to add and delete attribute options, and how to search for entries with attribute options.

**Example: Adding an Attribute Option by Using ldapmodify**

Suppose that you were adding the Spanish equivalent of an entry for John, and that the data for this user entry is in the `entry.ldif` file. This file contains the following:

```
dn: cn=john,c=us
changeType: modify
add: cn;lang-sp
cn;lang-sp: Juan
```

Issue this command to modify the file:

```
ldapmodify -p 389 -v -f entry.ldif
```

**Example: Deleting an Attribute Option by Using ldapmodify**

The following example deletes the `cn;lang-fr` attribute option from the entry for John. As in the previous example, assume that the data for this user entry is in the `entry.ldif` file. This file contains the following:

```
dn: cn=john,c=us
changeType: modify
delete: cn;lang-fr
```

Issue this command to modify the file:

```
ldapmodify -p 389 -v -f entry.ldif
```

**Example: Searching for Entries with Attribute Options by Using ldapsearch**

The following example retrieves entries with common name (cn) attributes that have an option specifying a language code attribute option. This particular example retrieves entries in which the common names are in French and begin with the letter R.

```
ldapsearch -p 389 -h myhost -b "c=US" -s sub "cn;lang-fr=R*"
```

Suppose that, in the entry for John, no value is set for the `cn;lang-it` language code attribute option. In this case, the following example fails:

```
ldapsearch -p 389 -h myhost -b "c=us" -s sub "cn;lang-it=Giovanni"
```

See Also: "Attribute Options" on page 3-11
Managing Knowledge References and Referrals

A **knowledge reference**, also called a **referral**, is represented in the directory as a particular type of **entry**. When you create a knowledge reference entry, you associate it with the **referral object class** and the **extensibleObject object class**. Typically, you create knowledge reference entries at the place in the **DIT** where you want to establish the partition.

A knowledge reference provides users with a referral containing an LDAP URL. You enter these URLs as values for the **ref** attribute. There can be multiple **ref** attributes specified for any knowledge reference entry. Similarly, there can be multiple knowledge reference entries in the **DIT**.

**See Also:** "Directory Partitioning" on page 3-18 for an overview of knowledge references and a description of a **smart knowledge reference** and a **default knowledge reference**.

This section contains these topics:

- Configuring Smart Referrals
- Configuring Default Referrals
- Client-Side Referral Caching

**Configuring Smart Referrals**

A search result can contain regular entries along with knowledge references. When a user performs a search operation, Oracle Internet Directory looks for the knowledge reference entry within the specified scope of the search. If it finds the knowledge reference, then Oracle Internet Directory returns a referral to the client.

If a user performs an add, delete, or modify operation on an entry located below the knowledge reference entry, then Oracle Internet Directory returns the referral.

For example, suppose you want to partition the DIT based on the geographical location of the directory servers. In this example, assume that:

- The **c=us** naming context is held locally on Server A and Server B in the United States.
- The **c=uk** naming context is held locally on Server C and Server D in the United Kingdom.

In this case, you would configure knowledge references between these two naming contexts as follows:

1. On Server A in the United States, configure a knowledge reference for the **c=uk** object on Server C and Server D:

   ```
   dn: c=uk
c: uk
ref: ldap://host C:389/c=uk
ref: ldap://host D:686/c=uk
objectclass: top
objectclass: referral
objectclass: extensibleObject
   ```

2. Configure a similar knowledge reference on Server C in the United Kingdom for the **c=us** object on Server A and Server B:

   ```
   dn: c=us
   ```
c: us
ref: ldap://host A:4000/c=us
ref: ldap://host B:5000/c=us
objectclass: top
objectclass: referral
objectClass: extensibleObject

Results:
- A client querying Server A with base o=foo,c=uk receives a referral.
- A client querying Server C with base o=foo,c=us receives a referral.
- An add operation of o=foo,c=uk on either Server A or Server B fails. Instead, Oracle Internet Directory returns a referral.

Configuring Default Referrals

Oracle Internet Directory uses the namingcontext attribute in the DSE to determine every directory naming context held locally by the server. Be sure that the namingContext attribute correctly reflects the naming context information.

You specify default referrals by entering a value for the ref attribute in the DSE entry. If the ref attribute is not in the DSE entry, then no default referral is returned.

When configuring a default referral, do not specify the DN in the LDAP URL.

For example, suppose that the DSE entry on Server A contains the following namingContext value:

namingcontext: c=us

Further, suppose that the default referral is:

Ref: ldap://host PQR:389

Now, suppose that a user enters an operation on Server A that has a base DN in the naming context c=canada, for example:

ou=marketing,o=foo,c=canada

This user would receive a referral to the host PQR. This is because Server A does not hold the c=canada base DN, and the namingcontext attribute in its DSE does not hold the value c=canada.

See Also: "Knowledge References and Referrals" on page 3-19 for a conceptual discussion of knowledge references

Client-Side Referral Caching

Referral caching is the process of storing referral information so that it can be easily accessed again and again. Suppose that a client queries Server A, which returns a referral to Server B. The client chases this referral and contacts Server B which performs the operation and returns the results to the client. Without referral caching, the next time the client makes the same query to Server A, the entire procedure is repeated, an unnecessary consumption of time and system resources.

However, if the referral information can be cached, then, in each subsequent query, the referral information can be obtained from cache and Server B can be contacted directly. This speeds up the operation.
Client-side referral caching enables each client to cache this referral information and use it to speed up referral processing.

**How Client-Side Referral Caching Works**

Referral entries are stored in a configuration file on the client. When a client establishes a session, it reads the referral information from this configuration file and stores them in a cache. This cache remains static, with no further updates being added during the session. From this point on, for every operation, the client looks up referral information in the cache.

The directory administrator prepares this configuration file for clients to use.

---

**Note:** The configuration file is optional for clients. If a file is not present, then client operations involving referrals still behave correctly. Thus it is not mandatory for administrator to prepare this file. The advantage of using the configuration file is that it speeds up the client/server operations involving referrals.

---

The configuration file consists of one or more referral sets. Each referral set consists of:

- The host name where a particular directory server is running
- One or more referral entries residing on that server

Each referral entry consists of a sequence of lines, each of which corresponds to one referral URL. The line separator is CR LF or LF.

```
ref_file=ref_file_content
ref_file_content=1*(referral_set)
referral_set=hostname    SEP    ref_entry_set   SEP
ref_entry_set=ref_entry     *(SEP    ref_entry)
ref_entry=1*(referralurl    SEP)
SEP=CR LF / LF
CR=0x0D
LF=0x0A
```

For example, consider two referral entries in a directory server running on host serverX:

```
dn: dc=acme, dc=com
ref: ldap://serverA:389/dc=acme, dc=com
ref: ldap://serverB:389/dc=acme, dc=com
```

```
dn: dc=oracle, dc=com
ref: ldap://serverC:389/dc=oracle, dc=com
ref: ldap://serverD:389/dc=oracle, dc=com
```

Consider the following referral entry in a directory server running on host serverY:

```
dn: dc=fiction, dc=com
ref: ldap://serverE:389/dc=fiction, dc=com
```

The corresponding referral.ora file looks like this:

```
ServerX
ldap://serverA:389/dc=acme, dc=com
ldap://serverB:389/dc=acme, dc=com
```

```
1ldap://serverC:389/dc=oracle, dc=com
1ldap://serverD:389/dc=oracle, dc=com
```
ServerY
ldap://serverE:389/dc=fiction, dc=com
As of 10g (10.1.4.0.1), the bulk tools have been rewritten as C executables. The new versions of the tools provide some new features:

- They reside in $ORACLE_HOME/ldap/bin, the same directory as other Oracle Internet Directory command-line tools.
- They have a consistent user interface.
- They can be invoked directly from the Windows command-line interface. You no longer need to install Cygwin or MKS Toolkit.
- Error and progress reporting have been enhanced.
- Each tool has its own log file in $ORACLE_HOME/ldap/log.

In addition to this chapter, bulk tools are also discussed in the following sections of this book:

- "Indexing an Attribute by Using Command-Line Tools" on page 11-15
- "Log File Locations" on page 14-1
- "Backing Up and Restoring a Small Directory or Specific Naming Context" on page 15-1
- "Migrating Data from LDAP-Compliant Directories" on page 27-2
- "Installing and Configuring Multimaster Replication" on page 30-5
- "Installing and Configuring One-Way or Two-Way LDAP-Based Replication" on page 30-19

**See Also:** The chapter on Oracle Internet Directory server administration tools in *Oracle Identity Management User Reference*.

**Note:** Stop all instances of Oracle Internet Directory before using `bulkload`. Before using any of the other bulk tools, either stop all instances of Oracle Internet Directory or disable the entry cache.

This chapter consists of the following sections:

- `bulkload`
- `bulkmodify`
- `bulkdelete`
- `ldifwrite`
bulkload

The bulk loader, bulkload, is a bulk management tool. It takes input data in LDIF or SQL*Loader format and loads that data directly into Oracle Internet Directory’s schema in the metadata repository. It has three main phases: check, generate and load.

In the check phase, bulkload parses and verifies the LDIF input data for the schema.

In the generate phase, bulkload generates intermediate files in SQL*Loader format.

In the load phase, bulkload can use either of two methods: bulk mode loading or incremental mode loading.

- When using bulk mode loading, bulkload loads generated intermediate files into the database. As it does so, it drops old indexes and generates new ones.
- When using incremental mode loading, bulkload loads intermediate files to tables in the database using insert mode. While it loads the data, bulkload updates the indexes.

Bulk mode loading is faster than incremental mode loading.

The bulk loader also supports the following features:

- It allows you to specify the number of threads in order to achieve parallelism during the generate and load phases.
- It has an encode option that allows you to use data in other languages.
- It has a restore option that allows you to retain the operational attributes specified in the LDIF file.
- It has an index option for index creation.
- It has a recover option that is useful for recovering from bulkload failures.
- When appending the data to existing directories, bulkload supports both bulk mode and incremental mode loading.
- The append option allows you to load data while the LDAP server is up and running.

At the beginning of the generate phase, the server’s orclServerMode is changed from read-write to read-modify. At the end of the generate phase, it is left in the read-modify state so that you cannot add entries to Oracle Internet Directory between the generate and load phases. This is necessary to maintain internal sequence numbers. You are expected to run the load phase immediately after the generate phase. At the end of the load phase, the servers’ orclServerMode is set back to read-write. Using bulkload with the recover option also sets orclServerMode back to read-write.

The bulkload tool generates the following output files in the $ORACLE_HOME/ldap/log directory:

- An output log, bulkload.log
- A list of duplicate DNs, duplicateDN.log
- Intermediate log files generated by the SQL*Loader, bsl_*.log
It generates the following output files in the $ORACLE_HOME/ldap/load directory:

- A list of bad LDIF entries, badentry.ldif
- A list of all dynamic group entries that can be added using ldapadd, dynGrp.ldif
- Intermediate files, *.ctl and *.dat

---

**Note:** If the applicable password policy has the `pwdmustchange` attribute set to true, then for every new entry loaded by `bulkload`, the `pwdreset` attribute is set to 1 by default. See Chapter 19, "Password Policies in Oracle Internet Directory" for more information.

---

**Note:** If you do not use the `bulkload` utility to populate the directory, then you must run the `oidstats.sql` tool to avoid significant search performance degradation.

---

**See Also:**

- The `oidstats.sql` command-line tool reference in Oracle Identity Management User Reference for a description and syntax for the `oidstats.sql` tool
- "Using Command-Line Tools" on page 5-8 for an overview of these tools

### `bulkload` Command Line Parameters

The `bulkload` tool accepts parameters in key=value format.

```
bulkload connect=connect_string
[ { [ check="TRUE" | "FALSE" ] [ restore="TRUE" | "FALSE" ] [ threads=num_of_threads ]
[ file=ldif_file ] } [ generate="TRUE" | "FALSE" ] [ append="TRUE" | "FALSE" ]
[ load="TRUE" | "FALSE" ] [ [ threads=num_of_threads ] file=ldif_file ]
[ index="TRUE" | "FALSE" ] [ [ threads=num_of_threads ] ]
[ [ recover="TRUE" | "FALSE" ] ]
[ [ verbose="TRUE" | "FALSE" ] ]
[ [ debug="TRUE" | "FALSE" ] ]
[ [ encode=character_set ] ]
```

Some of the parameter combinations are valid while others are invalid.

You must specify at least one of the following actions when you invoke `bulkload`: check, generate, load, append, recover, or index.

If check is TRUE, `bulkload` performs a schema check.

If generate is TRUE, `bulkload` generates intermediate files.

When using the check or generate action, you must specify the pathname to the LDIF data file.

If load is TRUE, `bulkload` loads intermediate files.

When append is TRUE, `bulkload` can perform its actions while the server is up and running.

Use the restore flag only when the LDIF file contains operational attributes, such as orclguid or creatorsname.

Do not specify recover with any other option.
The option combination check index verifies the existing indexes.

**Importing an LDIF File by Using bulkload**

To import an LDIF file, you use the `bulkload` utility. This section discusses the tasks to process an LDIF file through `bulkload`.

**See Also:** The `bulkload` command-line tool reference in *Oracle Identity Management User Reference*.

This section contains these topics:

- Task 1: Back Up the Oracle Database Server
- Task 2: Find Out the Oracle Internet Directory Password
- Task 3: Check Input for Schema and Data Consistency Violations and Generate the Input Files for SQL*Loader
- Task 4: Load the Input Files
- If Bulk Loading Fails

**Task 1: Back Up the Oracle Database Server**

Before you import the file, back up the Oracle database server as a safety precaution.

**See Also:** *Oracle Database Backup and Recovery Basics* in the Oracle Database Documentation Library

**Task 2: Find Out the Oracle Internet Directory Password**

To use `bulkload`, you must provide the Oracle Internet Directory password. The default password is `ods`, although the system administrator can change it by using the **OID Database Password Utility**.

**See Also:** The `oidpasswd` command-line tool reference in *Oracle Identity Management User Reference*.

**Task 3: Check Input for Schema and Data Consistency Violations and Generate the Input Files for SQL*Loader**

On UNIX, the `bulkload` tool usually resides in `$ORACLE_HOME/ldap/bin`. On Microsoft Windows, this tool usually resides in `ORACLE_HOME\ldap\bin`.

Check the input file and generate files for the SQL*Loader by typing:

```
bulkload connect="connect_string" check="TRUE" generate="TRUE" \
file="path_to_ldif-file_name"
```

All schema violations are reported in `$ORACLE_HOME/ldap/log/bulkload.log`. All bad entries are logged in `$ORACLE_HOME/ldap/load/badentry.ldif`. Use a text editor to fix all bad entries, then re-run bulkload with the check and generate options.

If there are duplicate entries, their DNs are logged in `$ORACLE_HOME/ldap/log/duplicateDN.log`. This is just for information purpose. The `bulkload` tool does not generate duplicate data for duplicate entries. It ignores duplicate entries.

**See Also:** *Oracle Internet Directory Administrator's Guide*
When bulkload completes successfully, it generates *.ctl and *.dat files in the $ORACLE_HOME/ldap/load directory to be used by SQL*Loader in load mode. Do not modify these files.

**Task 4: Load the Input Files**

After you have generated the input files, rerun bulkload with the load option. During this step, the *.dat files, which are in Oracle SQL*Loader specific format, are loaded into the database and the attribute indexes are created. The syntax is:

```sql
bulkload connect="connect_string" load="TRUE"
```

**If Bulk Loading Fails**

All loading errors are reported in the $ORACLE_HOME/ldap/log directory. They reside in bulkload.log and in the SQL*Loader-generated files *.bad and bsl_*.log.

If bulk load fails, the database might be in an inconsistent state. If this occurs, you should restore the database to its state prior to the bulkload operation.

You can do this either by using bulkload with the recover option or by restoring Oracle Internet Directory directory from a backup taken before you invoked bulkload.

**More bulkload Examples**

The following examples show some uses of bulkload.

**Example 1. Loading in Bulk Mode**

Typically, you load directory data just after installing Oracle Internet Directory. This requires three actions:

- Check the LDIF file for schema errors and generate the intermediate files
- Load the data into the Oracle Internet Directory metadata repository.

You perform these actions with command lines similar to these:

```sql
bulkload connect="conn_str" check="TRUE" generate="TRUE" file="LDIF_file"
bulkload connect="conn_str" load="TRUE"
```

You can omit the check phase if the LDIF data is from another Oracle Internet Directory node.

**Example 2. Loading in Incremental or Append Mode**

If you must add entries to an Oracle Internet Directory server that already contains data, and the server must be up and running at the same time, then you must use the incremental or append mode. This mode is usually faster than other methods of adding entries to the directory. However, you must ensure that the Oracle Internet Directory LDAP instances are in read-modify mode so that bulkload can append data.

You invoke bulkload in incremental or append mode with command lines similar to these:

```sql
bulkload connect="conn_str" check="TRUE" generate="TRUE" append="TRUE" \
file="LDIF_file"
bulkload connect="conn_str" load="TRUE" append="TRUE"
```
Example 3. Index Verification
The bulkload operation can either update indexes or create indexes. Sometimes, however, bulkload does not update or create the indexes properly. This is typically due to issues like improper sizing. If this happens, you can use bulkload to verify and re-create all the indexes.

Use the following syntax to invoke bulkload for verification of indexes:

```
bulkload connect="conn_str" check="TRUE" index="TRUE"
```

Example 4. Index Re-creation
To re-create indexes, use the following syntax:

```
bulkload connect="conn_str" index="TRUE"
```

Example 5. Data Recovery
The load phase of bulkload can fail because of issues like improper disk sizing. After such a failure, the directory data might be inconsistent. You can use the recover option to return the directory data to its pre-bulkload state. The syntax is:

```
bulkload connect="conn_str" recover="TRUE"
```

bulkmodify
The bulkmodify tool is useful for modifying the attributes of a large number of entries in an existing directory. It can perform add and replace operations on attribute values. It can operate on a naming context. Using filters, it can also operate selectively on a few entries under a specified naming context.

The bulkmodify tool does not allow add or replace operations on the following attributes:

- dn
- cn
- userpassword
- orclpassword
- orclentrylevelaci
- orclaci
- orclcertificatehash
- orclcertificatematch
- any binary attribute

It does not allow replace operation on the attribute objectclass.

It does not allow add for single-valued attributes.

Output from bulkmodify is logged in $ORACLE_HOME/ldap/log/bulkmodify.log.

See Also: The bulkmodify command-line tool reference in Oracle Identity Management User Reference

bulkmodify Command Line Parameters
Bulkmodify accepts parameters in key=value pairs.
Using Bulk Tools

bulkmodify connect=connect_string basedn=Base_DN
{[[add="TRUE"|"FALSE"]|[replace="TRUE"|"FALSE"]]} attribute=attribute_name
value=attribute_value [filter=filter_string] [size=transaction_size]
[threads=num_of_threads] [debug="TRUE"|"FALSE"] [encode=character_set]
[verbose="TRUE"|"FALSE"]

The number of threads should be from one to six times the number of CPUs.
Select either the add or the replace option. By default both are set to FALSE.

bulkmodify Usage Examples

The following examples show some uses of bulkmodify:

Example 1. Adding a Description for All Entries Under a Specified Naming Context
This example adds descriptions for all the entries under "c=us".
bulkmodify connect="connect_str" basedn="c=us" add="TRUE" \
  attribute="description" value="US citizen" filter="objectclass=*"

Example 2. Adding telephonenumber for All Entries Under a Specified Naming Context with the Same Manager
This example adds telephonenumber for all the entries under "c=us" that have the manager Anne Smith.
bulkmodify connect="connect_str" basedn="c=us" add="TRUE" \
  attribute="telephoneNumber" value="408-123-4567" filter="manager=Anne Smith"

Example 3. Replacing an Attribute for All Entries Under a Specified Naming Context
This example replaces pwdreset for all the entries under "c=us".
bulkmodify connect="connect_str" basedn="c=us" replace="TRUE" \
  attribute="pwdreset" value="1" filter="objectclass=*"

bulkdelete

Bulkdelete is useful for deleting the attributes of a large number of entries in an existing directory. Bulkdelete can delete entries specified under a naming context. By default, it deletes entries completely. It removes all traces of an entry from the database. If you use the option cleandb=FALSE, bulkdelete turns all entries into tombstone entries instead of deleting them completely.

Bulkdelete output is logged in $ORACLE_HOME/ldap/log/bulkdelete.log.

See Also: The bulkdelete command-line tool reference in Oracle Identity Management User Reference

bulkdelete Command Line Parameters

The bulkdelete tool accepts all parameters as key=value pairs.
bulkdelete connect=connect_string [[basedn=Base_DN]|[file=file_name]]
[cleandb="TRUE"|"FALSE"] [size=transaction_size] [encode=character_set]
[debug="TRUE"|"FALSE"] [threads=num_of_threads] [verbose="TRUE"|"FALSE"]

Select either the basedn or the file option. If cleandb is TRUE, bulkdelete removes entries completely from the database. By default cleandb is set to TRUE. The number of threads should be from one to six times the number of CPUs.
bulkdelete Usage Examples

The following examples demonstrate how to use bulkdelete.

Example 1. Delete All Entries Under a Specified Naming Context from Database
This example deletes all the entries under "c=us".

\[ \text{bulkdelete connect=connect\_str basedn=c=us cleandb=TRUE} \]

Example 2. Delete Entries Under a Naming Context and Make them Tombstone Entries
This example deletes all the entries under "c=us" and leaves them as tombstone entries.

\[ \text{bulkdelete connect=connect\_str basedn=c=us cleandb=FALSE} \]

Example 3. Delete Entries Under Specified Naming Contexts Given in File and Make them Tombstone Entries.
This example deletes all the entries under given basedn’s specified in file and leaves them as tombstone entries.

\[ \text{bulkdelete connect=connect\_str file=file cleandb=FALSE} \]

ldifwrite

The ldifwrite tool is used to dump data from an Oracle Internet Directory store to a file. Having the data in a file facilitates loading the data into another node for replication or backup storage. As it writes to the output file, the ldifwrite tool performs a subtree search, including all entries below the specified DN, as well as the DN itself. It dumps data in LDIF format. It can also dump entries under a specified replication agreement DN.

The ldifwrite tool can dump entries located by using specified filters.

Output from ldifwrite is logged in \$ORACLE_HOME/ldap/log/ldifwrite.log.

See Also: The ldifwrite command-line tool reference in Oracle Identity Management User Reference

Idifwrite Command Line Parameters

The ldifwrite tool accepts all parameters in keyword=value pair format.

\[ \text{ldifwrite connect=connect\_string basedn=Base\_DN ldiffile=LDIF\_Filename [filter=LDAP\_Filter] [threads=num\_of\_threads] [debug=TRUE|FALSE] [encode=character\_set] [verbose=TRUE|FALSE]} \]

Use the basedn option to specify the base DN or replication agreement DN.

The number of threads should be from one to six times the number of CPUs.

Idifwrite Usage Examples

The following examples show some uses of ldifwrite.
Example 1. Dumping All Entries Under a Specified Naming Context to an LDIF File

This example writes all the entries under "ou=Europe, o=imc, c=us" into the output.ldif file.

```bash
ldifwrite connect="connect_str" basedn="ou=Europe, o=imc, c=us" \
    ldiffile=output.ldif
```

The ldifwrite tool includes the operational attributes of each entry in the directory, including createtimestamp, creatorsname, and orclguid.

Example 2. Dumping Part of a Specified Naming Context to an LDIF File

This example uses the following naming context objects defined in partial replication:

dn: cn=includednamingcontext000001, cn=replication namecontext, orclagreementid=000001, orclreplicaid=node replica identifier, cn=replication configuration
orclincludednamingcontexts: c=us
orclexcludednamingcontexts: ou=Americas, c=us
orclexcludedattributes: userpassword
objectclass: top
objectclass: orclreplnamectxconfig

In this example, all entries under "c=us" are backed up except "ou=Americas,c=us". The userpassword attribute is also excluded. The command is:

```bash
ldifwrite connect="conn_str" \
    basedn="cn=includednamingcontext000001, cn=replication namecontext, orclagreementid=000001, orclreplicaid=node replica identifier, cn=replication configuration" ldiffile="ldif_file_name"
```

Example 3. Dumping Entries Under a Specified Naming Context to an LDIF File

This example writes all the entries that satisfy LDAP search filter criteria under "ou=Europe, o=imc, c=us" into the output.ldif file.

```bash
ldifwrite connect="connect_str" basedn="ou=Europe, o=imc, c=us" filter="uid=abc" \
    ldiffile="output.ldif"
```

**catalog**

The catalog tool is useful for creating indexes for or dropping indexes from existing attributes. The catalog tool makes an attribute searchable.

Output from catalog is logged in $ORACLE_HOME/ldap/log/catalog.log.

**catalog Command Line Parameters**

Catalog accepts all parameters in key=value pair format.

```bash
catalog connect=connect_string {[add="TRUE"|"FALSE"]|[delete="TRUE"|"FALSE"]} \
    {[attribute=attribute_name]|file=file_name]} {[logging="TRUE"|"FALSE"]} \
    {threads=num_of_threads} {[debug="TRUE"|"FALSE"]} {[verbose="TRUE"|"FALSE"]}
```

Select either the add or the delete option. By default both are set to FALSE.

The number of threads should be from one to six times the number of CPUs.

If logging is TRUE, catalog generates a redo log.
You can specify only one attribute argument on the command line at a time. To add or delete more than one attribute in a single command invocation, use the file option and specify a list of attributes in the file. Use one line per attribute, for example:

description
sn
title

catalog Usage Examples

The following examples show some uses of catalog:

Example 1. Changes a Searchable Attribute into a Non-searchable Attribute
This example drops an index from the attribute title.

catalog connect="connect_str" delete="TRUE" attribute="title"

Example 2. Changes a Non-searchable Attribute into a Searchable Attribute
This example adds an index to the attribute title.

catalog connect="connect_str" add="TRUE" attribute="title"

See Also:  "Indexing an Attribute by Using Command-Line Tools" on page 11-15
About Attribute Uniqueness

The attribute uniqueness feature prevents duplication of attribute values, both when adding and modifying them. For example, it prevents you from assigning to a new employee an identifier already assigned to another employee. Instead, the directory server terminates the operation and returns an error message.

You can define attribute uniqueness:

- Across the entire directory
  For example, to ensure that every entry in your directory that includes a `mail` attribute has a unique value for that attribute, you create an instance of attribute uniqueness associated with `mail`.

- Across one subtree for each attribute
  For example, suppose that MyCompany hosts the directories for SubscriberCompany1 and SubscriberCompany2. You can choose to enforce attribute uniqueness in SubscriberCompany1 only.

- Across one object class
  For example, suppose that `ID` is an attribute in both the `machine` object class and the `person` object class. If attribute uniqueness is enabled, then the directory server prevents you from adding either two machines or two people with the same `ID`. However, a `machine ID` attribute can have the same value as a `person ID` attribute.

To implement attribute uniqueness, you create an attribute uniqueness constraint entry in which you provide values for the attributes in Table 10-1 on page 10-2.
When you have created the entry and specified the attributes, before it performs an operation, the directory server:

- Uses the attribute uniqueness constraint to check all update operations
- Determines whether the operation applies to a monitored attribute, subtree, or object class

If an operation applies to a monitored attribute, suffix, or object class, and would cause two entries to have the same attribute value, then the directory server terminates the operation and returns a constraint violation error message to the client.

---

**Table 10–1 Attribute Uniqueness Constraint Entry**

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Mandatory?</th>
<th>Valid Value</th>
<th>Default Value</th>
<th>Default Effect</th>
</tr>
</thead>
<tbody>
<tr>
<td>orcluniqueattrname</td>
<td>Yes</td>
<td>Any string</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>orcluniquescope</td>
<td>No</td>
<td>One of the following:</td>
<td>sub</td>
<td>Searches the entire directory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- base: Searches the root entry only</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- onelevel: Searches one level only</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>- sub: Searches the entire directory</td>
<td></td>
<td></td>
</tr>
<tr>
<td>orcluniqueenable</td>
<td>No</td>
<td>Either 0 (disable) or 1 (enable)</td>
<td>0</td>
<td>Disables attribute uniqueness</td>
</tr>
<tr>
<td>orcluniquesubtree</td>
<td>No</td>
<td>Any string</td>
<td>&quot; &quot;</td>
<td>Searches the entire directory</td>
</tr>
<tr>
<td>orcluniqueobjectclass</td>
<td>No</td>
<td>Any string</td>
<td>&quot; &quot;</td>
<td>Searches all object classes</td>
</tr>
</tbody>
</table>

**Note:** The attribute uniqueness feature works on indexed attributes only.

---

**Rules for Creating Attribute Uniqueness**

This section describes and gives examples of rules you follow when creating attribute uniqueness constraints. It contains these topics:

- Specifying Multiple Attribute Names in an Attribute Uniqueness Constraint
- Specifying Multiple Subtrees in an Attribute Uniqueness Constraint
- Specifying Multiple Scopes in an Attribute Uniqueness Constraint
- Specifying Multiple Object Classes in an Attribute Uniqueness Constraint
- Specifying Multiple Subtrees, Scopes, and Object Classes in an Attribute Uniqueness Constraint

To understand the examples in this section, refer to Figure 10–1.
Specifying Multiple Attribute Names in an Attribute Uniqueness Constraint

When multiple attribute uniqueness constraints have different values in orcluniqueattrname, their effects are independent of each other.

For example, suppose that a user defines two attribute uniqueness constraints as follows:

Constraint1:
orcluniqueattrname: employee_id

Constraint2:
orcluniqueattrname: email_id

In this example, Constraint1 and Constraint2 enforce uniqueness on the specified attribute within their own attribute uniqueness scopes. Constraint1 and Constraint2 are independent of each other.

Specifying Multiple Subtrees in an Attribute Uniqueness Constraint

When multiple attribute uniqueness constraints have the same values in orcluniqueattrname, orcluniquescope and orcluniqueobjectclass, but different values in orcluniquesubtree, the union of subtree scopes specified by those attribute uniqueness constraints is checked.

For example, refer to Figure 10–1 on page 10-3. Suppose that a user defines two attribute uniqueness constraints as follows:

Constraint1:
orcluniqueattrname: employee_id
orcluniquesubtree: o=sales, c=us, cn=root
orcluniquescope: onelevel

Constraint2:
orcluniqueattrname: employee_id
orcluniquesubtree: o=hr, c=euro, cn=root
orcluniquescope: onelevel

In this example, the attribute uniqueness on employee_id is enforced against all entries under subtree o=sales, c=us, cn=root and o=hr, c=euro, cn=root—that
Rules for Creating Attribute Uniqueness

is, the directory server enforces the unique value of the employee_id attribute for employee3, employee4, employee7 and employee8.

Specifying Multiple Scopes in an Attribute Uniqueness Constraint

When multiple attribute uniqueness constraints have the same values in orcluniqueattrname, orcluniquesubtree and orcluniqueobjectclass, but different values in orcluniquescope, the attribute uniqueness constraint with the largest search scope takes effect.

For example, referring to Figure 10–1 on page 10-3, suppose that a user defines two attribute uniqueness constraints as follows:

Constraint1:
  orcluniqueattrname: employee_id
  orcluniquesubtree: c=us, cn=root
  orcluniquescope: onelevel

Constraint2:
  orcluniqueattrname: employee_id
  orcluniquesubtree: c=us, cn=root
  orcluniquescope: sub

In this example, the attribute uniqueness on employee_id is enforced against all entries under the subtree c=us,cn=root and the entry c=us,cn=root itself. Note that this is the same as if the user had defined only Constraint2.

Specifying Multiple Object Classes in an Attribute Uniqueness Constraint

When multiple attribute uniqueness constraints have the same values in orcluniqueattrname, orcluniquesubtree, and orcluniquescope, but different values in orcluniqueobjectclass, then the union of attributes belonging to those object classes is checked.

For example, look at Figure 10–1 on page 10-3. Suppose that a user defines two attribute uniqueness constraints as follows:

Constraint1:
  orcluniqueattrname: employee_id
  orcluniquesubtree: c=us, cn=root
  orcluniqueobjectclass: person

Constraint2:
  orcluniqueattrname: employee_id
  orcluniquesubtree: c=us, cn=root

In this example, the attribute uniqueness on employee_id is enforced against all entries under the subtree c=us, cn=root and the entry c=us, cn=root itself, no matter what object class those entries belong to. Note that Constraint2 specifies no orcluniqueobjectclass attribute, which is the same as specifying all object classes.
Specifying Multiple Subtrees, Scopes, and Object Classes in an Attribute Uniqueness Constraint

When multiple attribute uniqueness constraints have the same values in orcluniqueattrname, but different values in orcluniquesubtree, orcluniquescope, and orcluniqueobjectclass, the union of entries that belong to the attribute uniqueness scopes of different constraints are checked.

For example, referring to Figure 10-1 on page 10-3, suppose that a user defines two attribute uniqueness constraints as follows:

Constraint1:

- orcluniqueattrname: employee_id
- orcluniquesubtree: o=sales, c=us, cn=root
- orcluniquescope: onelevel
- orcluniqueobjectclass: person

Constraint2:

- orcluniqueattrname: employee_id
- orcluniquesubtree: c=euro, cn=root
- orcluniquescope: sub
- orcluniqueobjectclass: organization

In this example, the attribute uniqueness on employee_id is enforced against the following:

- All entries under the subtree o=sales, c=us, cn=root where their object class belongs to person
- All entries under subtree c=euro, cn=root and the entry c=euro, cn=root itself where their object class belongs to organization

Managing Attribute Uniqueness

This section contains these topics:

- Location of Attribute Uniqueness Entries
- Managing Attribute Uniqueness by Using Oracle Directory Manager
- Managing Attribute Uniqueness by Using Command-Line Tools

Location of Attribute Uniqueness Entries

Attribute uniqueness constraint entries are stored under cn=unique,cn=Common,cn=Products,cn=OracleContext.

Managing Attribute Uniqueness by Using Oracle Directory Manager

You can use Oracle Directory Manager to create, modify, and delete attribute uniqueness constraint entries.

Creating an Attribute Uniqueness Constraint Entry

1. In the navigator pane, expand in succession Oracle Internet Directory Servers, directory server instance, and Attribute Uniqueness Management. The Attribute Uniqueness Management window displays a list of existing attribute uniqueness constraint entries in the right pane.
2. On the toolbar, choose Create. This displays the New Constraint window.
   In the New Constraint dialog box, enter values for the fields. These are described in Table A–8 on page A-5.

3. Choose OK. This returns you to the Attribute Uniqueness Management window. The entry you just created appears in the list of attribute uniqueness constraint entries.

4. Choose Apply.

**Modifying an Attribute Uniqueness Constraint Entry by Using Oracle Directory Manager**

To modify an attribute uniqueness constraint entry:

1. In the navigator pane, expand in succession Oracle Internet Directory Servers, directory server instance, and Attribute Uniqueness Management. The Attribute Uniqueness Management window displays a list of existing attribute uniqueness constraint entries in the right pane.

2. In the Attribute Uniqueness Management window, select the attribute uniqueness constraint entry you want to modify, then choose Edit. The Attribute Uniqueness Constraint window for that attribute appears.

3. In the Attribute Uniqueness Constraint window, enter your modifications in the appropriate fields, then choose OK. This returns you to the Attribute Uniqueness Management window.

4. Choose Apply.

**Deleting an Attribute Uniqueness Constraint Policy by Using Oracle Directory Manager**

To delete an attribute uniqueness constraint policy:

1. In the navigator pane, expand in succession Oracle Internet Directory Servers, directory server instance, and Attribute Uniqueness Management. The Attribute Uniqueness Management window displays a list of existing attribute uniqueness constraint entries in the right pane.

2. In the Attribute Uniqueness Management window, select the attribute uniqueness constraint entry you want to delete, then choose Edit. The Attribute Uniqueness Constraint window for this attribute appears.

3. Choose Delete, then, when prompted, confirm the deletion. This returns you to the Attribute Uniqueness Constraint window. The entry you deleted no longer appears in the list of attribute uniqueness constraint entries.

**Managing Attribute Uniqueness by Using Command-Line Tools**

This section contains these topics:

- Enabling and Disabling Attribute Uniqueness by Using Command-Line Tools
- Creating Attribute Uniqueness Constraint Entries by Using Command-Line Tools
- Modifying Attribute Uniqueness Constraint Entries by Using Command-Line Tools
- Deleting Attribute Uniqueness Constraint Entries by Using Command-Line Tools
Enabling and Disabling Attribute Uniqueness by Using Command-Line Tools
You can enable or disable attribute uniqueness for an existing attribute uniqueness constraint entry.

To enable attribute uniqueness for an existing attribute uniqueness constraint entry:
1. Set the orcluniqueenable attribute to 1 by using ldapmodify.
2. Restart the directory server to enable the policy.

To disable attribute uniqueness:
1. Set the orcluniqueenable attribute to 0 by using ldapmodify.
2. Restart the directory server to disable the policy.

Creating Attribute Uniqueness Constraint Entries by Using Command-Line Tools
To enable attribute uniqueness, specify an attribute uniqueness constraint entry with the attributes listed in Table 10–1 on page 10-2.

Creating Attribute Uniqueness Across an Entire Directory by Using Command-Line Tools
To create an instance of attribute uniqueness across an entire directory, specify an attribute name for which you want to enforce value uniqueness.

For example, to make employee identifiers unique for all US employees at MyCompany, you would follow these steps:
1. Create an attribute uniqueness constraint entry (in LDIF format) as follows:
   ```ldif
dn: cn=constraint1, cn=unique, cn=common, cn=products, cn=oraclecontext
objectclass: orclUniqueConfig
orcluniqueattrname: employeenumber
orcluniquesubtree: o=MyCompany, c=US
orcluniqueobjectclass: person
```
2. Apply the attribute uniqueness feature by loading the attribute uniqueness constraint entry as follows:
   ```bash
   ldapadd -h host -p port -D DN -w password -f constraint1.dat
   ```
3. Restart the directory server.

Note: To specify a uniqueness constraint for the orclcommonusername attribute, use the LDIF template file:
```
$ORACLE_HOME/ldap/schema/oid/uniquenessConstraint.ldif
```

Creating Attribute Uniqueness Across One Subtree by Using Command-Line Tools
To create an instance of attribute uniqueness across one or more subtrees, specify:
- An attribute name for which you want to enforce value uniqueness
- Subtree locations under which you want the uniqueness constraint to be enforced

For example, suppose that MyCompany hosts the directories for SubscriberCompany1 and SubscriberCompany2, and you want to enforce the uniqueness of the employee identifier attribute in SubscriberCompany1 only. When you add an entry such as uid=dlin,ou=people,o=SubscriberCompany1,dc=MyCompany, dc=com, you must enforce uniqueness only in the
Managing Attribute Uniqueness

o=SubscriberCompany1, dc=MyCompany, dc=com subtree. Do this by listing the DN of the subtree explicitly in the attribute uniqueness constraint configuration.

In this case, the LDIF file would look like this:

dn: cn=constraint1, cn=unique, cn=common, cn=products, cn=oraclecontext
objectclass: orclUniqueConfig
orcluniqueattrname: employeenumber
orcluniquesubtree: o=SubscriberCompany1, dc=MyCompany, dc=com

Creating Attribute Uniqueness Across One Object Class by Using Command-Line Tools

To create an instance of attribute uniqueness across one object class, specify:

- An attribute name for which you want to enforce value uniqueness
- Object class name

In this case, the LDIF file would look like this:

dn: cn=constraint1, cn=unique, cn=common, cn=products, cn=oraclecontext
objectclass: orclUniqueConfig
orcluniqueattrname: employeenumber
orcleuniqueobjectclass: person

Modifying Attribute Uniqueness Constraint Entries by Using Command-Line Tools

To modify an attribute uniqueness entry, use create an LDIF file for the entry, then use ldapmodify to upload it into the directory.

For example, suppose there is an existing attribute uniqueness constraint entry:

dn: cn=constraint1, cn=unique, cn=common, cn=products, cn=oraclecontext
objectclass: orclUniqueConfig
orcluniqueattrname: employeenumber
orcluniquesubtree: o=MyCompany, c=US
orcleuniqueobjectclass: person

To enforce the constraint against c=US, instead of o=MyCompany, you would perform these steps:

1. Create an LDIF entry to change the orcluniquenesssubtree:

   dn: cn=constraint1, cn=unique, cn=common, cn=products, cn=oraclecontext
   changetype: modify
   replace: orcluniquesubtree
   orcluniquesubtree: o=Oracle Corporation, c=US

2. Use ldapmodify to apply the change to directory server.

   ldapmodify -p port -D user -w password -f file_name

3. Restart the directory server to effect this change.

Deleting Attribute Uniqueness Constraint Entries by Using Command-Line Tools

Use the ldapdelete command-line tool to delete an attribute uniqueness constraint policy.

1. Remove the attribute uniqueness constraint entry from the directory by using ldapdelete.

   ldapdelete -p port -D bind_DN -w password """"cn=constraint1,cn=unique,cn=common,cn=products,cn=oraclecontext"

2. Restart the directory server to effect this change.
Limitations of Attribute Uniqueness in Oracle Internet Directory 10g (10.1.4.0.1)

When an attribute uniqueness constraint is present in the Oracle Internet Directory replication environment, be careful about configuring the attribute uniqueness constraints on each server. This section contains these topics:

- Simple Replication Scenario
- Multimaster Replication Scenario

Simple Replication Scenario
Because all modifications by client applications are performed on the supplier server, the attribute uniqueness constraint should be enabled on that server. It is not necessary to enable the attribute uniqueness constraint on the consumer server. Enabling the attribute uniqueness constraint on the consumer server does not prevent the directory server from operating correctly, but it can cause a performance degradation.

Multimaster Replication Scenario
In a multimaster replication scenario, nodes serve as both suppliers and consumers of the same replica. Multimaster replication uses a loosely consistent replication model.

Enabling an attribute uniqueness constraint on one of the servers does not ensure that attribute values are unique across both masters at any given time. Enabling an attribute uniqueness constraint on only one server can cause inconsistencies in the data held on each replica.

The attribute uniqueness constraint must be enabled on both masters. However, there may still be an inconsistent state. For example, in both masters we can successfully modify entries to the same attribute value. However, when the changes are later replicated to the other node, the conflict becomes apparent. You must take this type of conflict resolution into consideration as well, deciding whether conflict resolution should be the replication server’s responsibility.
This chapter explains how to administer the Oracle Internet Directory object classes and attributes.

This chapter contains these topics:
- About the Directory Schema
- Object Classes in the Directory
- Attributes in the Directory
- How to Extend the Number of Attributes Associated with Entries
- Attribute Aliases In the Directory
- Matching Rules in the Directory
- Syntaxes in the Directory

About the Directory Schema

A directory schema:
- Contains rules about the kinds of objects you can store in the directory
- Contains rules for how directory servers and clients treat information during operations such as a search
- Helps to maintain the integrity and quality of the data stored in the directory
- Reduces duplication of data
- Provides a predictable way for directory-enabled applications to access and modify directory objects

The directory schema contains all information about how data is organized in the DIT—that is, metadata such as that for an object class, an attribute, a matching rule, and syntax. This information is stored in a special class of entry called a subentry. More specifically, Oracle Internet Directory, following LDAP Version 3 standards, stores this information in subentries of type subSchemaSubentry.

You can add new object classes and attribute types to a subentry of type subSchemaSubentry. You cannot add new matching rules and syntaxes beyond those already supported by Oracle Internet Directory.

Prior to 10g (10.1.4.0.1) there was only one subSchemaSubentry, directly under the root DSE entry, called cn=subSchemaSubentry, and you always added object classes and attribute types directly to it. As of 10g (10.1.4.0.1), the entry cn=subSchemaSubentry can now have subordinate entries. Any application using...
Oracle Internet Directory that needs to add schema components can create its own subSchemaSubentry under cn=subSchemaSubentry and add the schema components to it. Figure 11–1 shows some subSchemaSubentry entries that were defined by applications, such as cn=userdefined1,cn=subschemasubentry and cn=userdefined2,cn=subschemasubentry.

All the schema entries are listed in the root DSE attribute subschemasubentry.

You cannot use bulkload to add subSchemaSubentry entries. You must use ldapadd.

Object Classes in the Directory

This section contains these topics:

- About Object Class Management
- Guidelines for Adding, Modifying, and Deleting Object Classes
- Managing Object Classes by Using Oracle Directory Manager
- Managing Object Classes by Using Command-Line Tools

About Object Class Management

This section explains how to add and modify an object class. Oracle recommends that you understand the basic concepts of directory components before attempting to add to or modify the base schema in the directory.

When you add an entry, you associate it with one or more object classes. Each object class contains attributes that you want to associate with the new entry. For example, if you are creating an entry for an employee, you can associate it with the person object class. This object class contains many of the attributes that you want to associate with that employee entry, including, for example, name, address, and telephone number.

Inheritance

Each object class derives from a hierarchy of superclasses, and it inherits attributes from these superclasses. By default, all object classes inherit from the top object class. When you assign an object class to an entry, the entry inherits all of the attributes of both that object class as well as its superclasses.

Mandatory and Optional Attributes in Object Classes

The attributes that entries inherit from a super class may be either mandatory or optional. Values for optional attributes need not be present in the directory entry.
You can specify for any object class whether an attribute is mandatory or optional; however, the characteristic you specify is binding only for that object class. If you place the attribute in another object class, you can again specify whether the attribute is mandatory or optional for that object class. You can:

- Add a new, non-standard object class and assign it existing attributes
- Select from existing standard object classes
- Modify an existing object class, assigning it a different set of attributes
- Add and modify existing attributes

See Also:

- "About Attribute Management" on page 11-9
- "Object Classes" on page 3-12 for a conceptual overview of object classes
- "LDAP Schema Overview" in Oracle Identity Management User Reference for a list of schema elements installed with Oracle Internet Directory

Addition of Entries in Top-Down Sequence

Entries must be added in a top-down sequence—that is, when you add an entry, all of its parent entries must already exist in the directory. Similarly, when you add entries that reference object classes and attributes, those referenced object classes and attributes must already exist in the directory schema. In most cases this will not be a problem because the directory server is delivered with a full set of standard directory objects.

Object Class Explosion

When you add or perform an operation on an entry, you do not need to specify the entire hierarchy of superclasses associated with that entry. You can specify only the leaf object classes. Oracle Internet Directory resolves the hierarchy for the leaf object classes and enforces the information model constraints. For example, the inetOrgPerson object class has top, person and organizationalPerson as its superclasses. When you create an entry for a person, you need to specify only inetOrgPerson as the object class. Oracle Internet Directory then enforces the schema constraints defined by the respective superclasses, namely, top, person, and organizationalPerson.

Guidelines for Adding, Modifying, and Deleting Object Classes

This section tells you what to keep in mind when adding, modifying, or deleting object classes.

Guidelines for Adding Object Classes

When you add object classes, keep the following in mind:

- Every structural object class must have top as a superclass.
The name and the object identifier of an object class must be unique across all the schema components. The actual attribute name and any attribute name aliases must be unique across all attribute names and attribute aliases. The Object Identifier must begin with the unique identifier 2.16.840.1.113894 followed by either the Oracle-supplied prefix .9999 or a site-specific prefix.

- Schema components referred to in the object class, such as superclasses, must already exist.
- The superclass of an abstract object class must be abstract also.
- It is possible to redefine mandatory attributes in a superclass into optional attributes in the new object class. Conversely, optional attributes in a superclass can be redefined into mandatory attributes in the new object class.

**Note:** Every schema object in the Oracle Internet Directory has certain limitations. For example, some objects cannot be changed. These limitations are explained as constraints and rules in this chapter.

**See Also:** "Subclasses, Superclasses, and Inheritance" on page 3-12 for a conceptual discussion of these terms

**Guidelines for Modifying Object Classes**

This section discusses the types of modifications you can make to an existing object class. You can perform modifications through Oracle Directory Manager and through the command-line tools.

You can make these changes to an object class:

- Change a mandatory attribute into an optional attribute
- Add optional attributes
- Add additional superclasses
- Convert abstract object classes into structural or auxiliary object classes unless the abstract object class is a superclass to another abstract object class

When you modify object classes, keep these guidelines in mind:

- You cannot modify an object class that is part of the standard LDAP schema. You can, however, modify user-defined object classes.
- If existing object classes do not have the attributes you need, you can create an auxiliary object class and associate the needed attributes with that object class.
- You cannot add additional mandatory attributes to an existing object class.
- You cannot modify object classes in the base schema.
- You cannot remove attributes or superclasses from an existing object class.
- You cannot convert structural object classes to other object class types.
- You should not modify an object class if there are entries already associated with it.
Guidelines for Deleting Object Classes
There are also some limitations on deleting object classes:

- You cannot delete object classes from the base schema.
- You can delete object classes that are not in the base schema as long as they are not directly or indirectly referenced by other schema components. For example, there may be some directory entries referring to these object classes. Deleting these object classes renders these entries inaccessible.

Managing Object Classes by Using Oracle Directory Manager
This section tells you how to use Oracle Directory Manager to search for object classes, view their properties, add, modify, and delete them.

Searching for Object Classes by Using Oracle Directory Manager
You can specify your search for an object class by:

- Selecting an object class property, for example, a name or an object identifier
- Entering a value for the property you selected
- Selecting a search filter specifying the relationship between the object class property you selected and the value you entered, for example, Begins With or Exactly Matches

This section provides more details on how to enter an object class search.

To search for an object class:

1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance.
3. In the right pane, choose Find Object Classes. The Find: Object Classes dialog box appears.
4. On the search criteria bar, from the menu farthest to the left, select the property of the object class you want to search for. Options are listed and described in Table A–28 on page A-17.

Note: Not all attributes are used in every object class. Be sure that the attribute you specify actually corresponds to one in the object class for which you are looking. Otherwise, the search will fail.

5. In the menu in the middle of the search criteria bar, select the filter you want to use for your search. Options are listed and described in Table A–29 on page A-18.
6. In the text box at the right end of the search criteria bar, type the value of the property of the object class you are searching for. For example, to search for all object classes with names that begin with the letters `orcl`, type those letters in the text box at the right end of the search criteria bar.

7. Below the Criteria field are five buttons described in the next table. Use these buttons to further refine your search.

8. Choose Search. The results of your search appear in the window at the lower portion of the Find:Object Class dialog box.

Viewing Properties of Object Classes by Using Oracle Directory Manager
To view all object classes in the schema:

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

2. Select Schema Management.

3. In the right pane, select the Object Classes tab page.

   To examine an individual object class and its attributes, in the Object Classes tab page, choose the object class. The properties of the selected object class appear in the Object Class dialog box.

Adding Object Classes by Using Oracle Directory Manager
To add object classes by using Oracle Directory Manager:

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

2. Select Schema Management.

3. In the right pane, select the Object Classes tab and, in the toolbar, choose Create. The New Object Class dialog box appears.

   Alternatively, in the Object Classes tab page, select an object class that is similar to one you would like to create, and then choose Create Like. The New Object Class dialog box displays the attributes of the selected object class. You can create the new object class by using this one as a template.

4. In the New Object Class dialog box, enter the information in the fields. These are described in Table A–31 on page A-19.

5. Choose OK.

See Also:

- "Object Class Types" on page 3-12
- "Subclasses, Superclasses, and Inheritance" on page 3-12
- Oracle Directory Manager online help for further details about adding object classes

Modifying Object Classes by Using Oracle Directory Manager
To modify an object class:

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

2. Select Schema Management.
3. In the right pane, select the **Object Classes** tab and choose the object class you want to modify. The Object Class dialog box appears.

4. In the Object Class dialog box, modify or add the information in the fields. These are described in **Table A-31 on page A-19**.

5. Choose **OK**.

**See Also:**
- "Object Class Types" on page 3-12
- "Subclasses, Superclasses, and Inheritance" on page 3-12

---

**Note:** You can add attributes to an auxiliary object class or a user-defined structural object class.

**See Also:** Example: Adding a New Attribute to an Auxiliary or User-Defined Object Class on page 11-8 for an example of adding attributes to an auxiliary object class

---

**Deleting Object Classes by Using Oracle Directory Manager**

**Caution:** Oracle recommends that you not delete object classes from the base schema. If you delete an object class that is referenced by any entries, those entries then become inaccessible.

Should you decide to delete an object class from the base schema, be careful not to delete one that is in use or that you might want to use in the future.

To delete an object class by using Oracle Directory Manager:

1. In the navigator pane, select **Schema Management**.
2. In the right pane, select the **Object Classes** tab page and select the object class you want to delete.
3. Choose **Delete**.

**Managing Object Classes by Using Command-Line Tools**

You can use command-line tools to add or modify existing object classes in the directory schema. The command-line tools enable you to use input files. Furthermore, the commands can be batched together in scripts.

To add or modify schema components, use `ldapmodify`.

**See Also:** The `ldapmodify` command-line tool reference in *Oracle Identity Management User Reference*

**Example: Adding a New Object Class**

In this example, an LDIF input file, `new_object_class.ldif`, contains data similar to this:

dn: cn=subschemasubentry
changetype: modify
add: objectclasses
Attributes in the Directory

This section contains these topics:

objectclasses: ( 2.16.840.1.113894.9999.12345 NAME 'myobjclass' SUP top STRUCTURAL MUST ( cn $ sn ) MAY ( telephonenumber $ givenname $ myattr ) )

Be sure to leave the mandatory space between the opening and closing parentheses and the object identifier.

To load the file, enter this command:

ldapmodify -h myhost -p 389 -f new_object_class.ldi

This example:

- Adds the structural object class named myobjclass
- Gives it an object identifier of 2.16.840.1.113894.9999.12345.
- Specifies top as its superclass
- Specifies cn and sn as mandatory attributes
- Allows telephonenumber, givenname, and myattr as optional attributes

Note that all the attributes mentioned must exist prior to the execution of the command.

To create an abstract object class, follow the previous example, replacing the word STRUCTURAL with the word ABSTRACT.

Example: Adding a New Attribute to an Auxiliary or User-Defined Object Class

To add a new attribute to either an auxiliary object class or a user-defined structural object class, use ldapmodify. This example deletes the old object class definition and adds the new definition in a compound modify operation. The change is committed by the directory server in one transaction. Existing data is not affected. The input file should be as follows:

dn: cn=subschemasubentry
changetype: modify
delete: objectclasses
objectclasses: old value
- add: objectclasses
objectclasses: new value

For example, to add the attribute changes to the existing object class country, the input file would be:

dn: cn=subschemasubentry
changetype: modify
delete: objectclasses
objectclasses:
 ( 2.16.840.1.113894.9999.12345 NAME 'country' SUP top STRUCTURAL MUST c MAY
   ( searchGuide $ description  ) )
- add: objectclasses
objectclasses:
 ( 2.16.840.1.113894.9999.12345 NAME 'country' SUP top STRUCTURAL MUST c MAY
   ( searchGuide $ description $ changes ) )
About Attribute Management

You need to understand attributes from a conceptual standpoint before attempting operations involving attributes.

In most cases, the attributes available in the base schema will suit the needs of your organization. However, if you decide to use an attribute not in the base schema, you can add a new attribute or modify an existing one.

By default, attributes are multivalued. You can specify an attribute as single-valued by using either Oracle Directory Manager or command-line tools.

Rules for Adding Attributes
The rules for adding attributes are:

- The name and the object identifier of an attribute must be unique across all the schema components.
- Syntax and matching rules must agree.
- Any super attributes must already exist.
- The length of an attribute name must not exceed 127 characters.

Rules for Modifying Attributes
The rules for modifying attributes are:

- The name and the object identifier of an attribute must be unique across all the schema components.
- The syntax of an attribute cannot be modified.
- A single-valued attribute can be made multi-valued, but a multi-valued attribute cannot be made single-valued.
- You cannot modify or delete base schema attributes.

See Also:
- "Attributes" on page 3-8 for a conceptual discussion of attributes
Rules for Deleting Attributes
The rules for deleting attributes are:

- You can delete only user-defined attributes. Do not delete attributes from the base schema.
- You can delete any attribute that is not referenced directly or indirectly by some other schema component.

If you delete an attribute that is referenced by any entry, that entry will no longer be available for directory operations.

See Also:  "About LDAP Attribute Syntax" in Oracle Identity Management User Reference for information about using syntax to specify the size of the attribute value

Managing Attributes by Using Oracle Directory Manager
This section tells you how to use Oracle Directory Manager to search for, view, add, modify, delete, and index attributes.

Viewing All Directory Attributes by Using Oracle Directory Manager
To view attributes by using Oracle Directory Manager:

1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance.
2. Select Schema Management.
3. In the right pane, select the Attributes tab page. This tab page displays a table containing the attribute properties. The columns in this table are described in Table A–32 on page A-19.

See Also:  "Viewing Attributes for a Specific Entry by Using Oracle Directory Manager" on page 8-3 for instructions about how to view attributes for a specific entry.

Searching for Attributes by Using Oracle Directory Manager
To search for attributes by using Oracle Directory Manager:

1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance.
3. Select the Attributes tab page.
4. Choose the Find Attributes button in the lower right corner. The Find Attributes dialog box appears.
5. In the menu at the left end of the search criteria bar, select the property of the attributes for which you want to search. Options are described in Table A–32 on page A-19.
6. In the menu in the middle of the search criteria bar, select the filter you want to use for your search. Options are described in Table A–33 on page A-20.
7. In the text box at the right end of the search criteria bar, type part or all of the value of the attribute for which you want to search. For example, to search for all attributes whose names begin with the letters orcl, you would type those letters
in the text box at the right end of the search criteria bar and create the phrase Name Begins With orcl.

8. To further refine your search, use the buttons in the Search Criteria box to enhance the search criteria bar. These are described in Table A–34 on page A-20.

9. Choose Search. The results of your search appear in the window at the lower portion of the Find Attributes dialog box.

Adding an Attribute by Using Oracle Directory Manager
You can add a completely new attribute, or copy from an existing one.

   Tip: Because equality, syntax, and matching rules are numerous and complex, it may be simpler to copy these characteristics from a similar existing attribute. See "Creating a New Attribute from an Existing One by Using Oracle Directory Manager" on page 11-11.

Adding a New Attribute by Using Oracle Directory Manager  To add a new attribute:

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

2. Select Schema Management.

3. In the right pane, select the Attributes tab, then choose the Create button in the toolbar. The New Attribute Type dialog box appears. It contains two tab pages—General and Advanced—with fields in which you either enter values or select from menus.

4. In the General tab, enter values in each of the fields. These are described in Table A–35 on page A-21.

5. Select the Advanced tab, and enter values in each of the fields. These are described in Table A–36 on page A-21.

6. Choose OK.

Note: To use this attribute, remember to declare it to be part of the attribute set for an object class. You do this by selecting Schema Management in the navigator pane, then, in the right pane, selecting the Object Classes tab page. For further instructions, see "Guidelines for Modifying Object Classes" on page 11-4.

Creating a New Attribute from an Existing One by Using Oracle Directory Manager  To add an attribute by copying an existing attribute:

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

2. Select Schema Management.

3. In the right pane, select the Attributes tab.

4. In the Attributes tab page, select the attribute you want to copy.

5. Choose Create Like. The New Attribute Type dialog box for that attribute appears. This dialog box contains two tab pages—General and Advanced.
6. Select the **General** tab and enter values in each of the fields. These are described in Table A–35 on page A-21. You must always change the DN to that of the new attribute.

7. Select the **Advanced** tab and enter values in each of the fields. These are described in Table A–36 on page A-21.

8. Choose OK.

---

**Note:** To use this attribute, remember to declare it to be part of the attribute set for an object class. You do this by selecting Schema Management in the navigator pane, then, in the right pane, selecting the Object Classes tab page. For further instructions, see “Guidelines for Modifying Object Classes” on page 11-4.

---

**Modifying an Attribute by Using Oracle Directory Manager**

To modify an attribute by using Oracle Directory Manager:

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

2. Select **Schema Management**.

3. In the right pane, select the **Attributes** tab, then select an editable attribute in the list.

4. Choose **Edit**. The Attribute dialog box displays two tab pages—**General** and **Advanced**—with fields in which you enter values either by typing or selecting from menus.

5. Select the **General** tab and enter values in each of the fields. These are described in Table A–35 on page A-21.

6. Select the **Advanced** tab and enter values in each of the fields. These are described in Table A–36 on page A-21.

7. Choose OK.

**Deleting an Attribute by Using Oracle Directory Manager**

---

**Note:** You can delete only user-defined attributes. Do not delete attributes from the base schema.

---

To delete an attribute:

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

2. Select **Schema Management**.

3. In the right pane, select the **Attributes** tab, then select an editable attribute in the list.

4. Choose **Delete**.
Indexing an Attribute by Using Oracle Directory Manager

Oracle Internet Directory uses indexes to make attributes available for searches. When Oracle Internet Directory is installed, certain attributes are already indexed. If you want to use additional attributes in search filters, you must index them.

---

**Note:** You can use Oracle Directory Manager to index an attribute only at the time when you create it. You cannot use Oracle Directory Manager to index an already existing attribute. To index an already existing attribute, use the Catalog Management tool as described in "Indexing an Attribute by Using Command-Line Tools" on page 11-15.

You can index only those attributes that have:

- An equality matching rule
- Matching rules supported by Oracle Internet Directory as listed in "About LDAP Attribute Matching Rules" in Oracle Identity Management User Reference
- Less than 128 characters in their names

---

Viewing Indexed Attributes by Using Oracle Directory Manager To view indexed attributes:

1. In the navigator pane, expand **Oracle Internet Directory Servers**, then **directory server instance**.
2. Select **Schema Management**.
3. In the right pane, select the **Attributes** tab page. This tab page displays all of the attributes in the schema. A selected check box in the Indexed column indicates an indexed attribute.

---

Adding an Index to an Attribute by Using Oracle Directory Manager To add an index to an attribute:

1. Create an attribute as described in "Adding an Attribute by Using Oracle Directory Manager" on page 11-11.
2. In the New Attribute Type dialog box, on the **Advanced** tab page, select the **Indexed** check box.

---

Dropping an Index from an Attribute by Using Oracle Directory Manager To drop an index from an attribute:

1. In the navigator pane, expand **Oracle Internet Directory Servers**, then **directory server instance**.
2. Select **Schema Management**.
3. In the right pane, select the **Attributes** tab.
4. Select the indexed attribute. Note that this must be an attribute that is editable as indicated by the icon to the left of the attribute name.
5. Choose **Drop Index**.

---

Managing Attributes by Using Command-Line Tools

This section discusses adding, modifying, and indexing attributes by using command-line tools.
**Adding and Modifying Attributes by Using ldapmodify**

To add a new attribute to the schema by using ldapmodify, type a command similar to the following at the system prompt:

```
ldapmodify -h host -p port -f ldif_file_name
```

The LDIF file contains data similar to this:

```
dn: cn=subschemasubentry
changetype: modify
add: attributetypes
attributetypes: ( 1.2.3.4.5 NAME 'myattr' SYNTAX '1.3.6.1.4.1.1466.115.121.1.38' )
```

To specify an attribute as single-valued, include in the attribute definition entry in the LDIF file the keyword SINGLE-VALUE with surrounding white space.

You can find a given syntax Object ID by using either Oracle Directory Manager or the ldapsearch command line tool.

**See Also:**

- The `ldapmodify` command-line tool reference in *Oracle Identity Management User Reference* for a detailed explanation of `ldapmodify` and its options
- “Syntaxes in the Directory” on page 11-26 for instructions on how to view syntaxes by using either Oracle Directory Manager or ldapsearch

**Deleting Attributes by Using ldapmodify**

```
ldapmodify -h host -p port -f ldif_file_name
```

The LDIF file contains data similar to this:

```
dn: cn=subschemasubentry
changetype: modify
delete: attributetypes
attributetypes: ( 1.2.3.4.5 NAME 'myattr' SYNTAX '1.3.6.1.4.1.1466.115.121.1.38' )
```

You can find a given syntax Object ID by using either Oracle Directory Manager or the ldapsearch command line tool.

**Note:** You can delete only user-defined attributes. Do not delete attributes from the base schema.
See Also:

- The `ldapmodify` command-line tool reference in *Oracle Identity Management User Reference* for a detailed explanation of `ldapmodify` and its options
- “Syntaxes in the Directory” on page 11-26 for instructions on how to view syntaxes by using either Oracle Directory Manager or `ldapsearch`

### Indexing an Attribute by Using Command-Line Tools

Oracle Internet Directory uses indexes to make attributes available for searches. When Oracle Internet Directory is installed, the entry `cn=catalogs` lists available attributes that can be used in a search.

If you want to use additional attributes in search filters, you must add them to the catalog entry. You can index only those attributes that have:

- An equality matching rule
- Matching rules supported by Oracle Internet Directory as listed in "About LDAP Attribute Matching Rules" in *Oracle Identity Management User Reference*
- No more than 128 characters in their names

You can index a new attribute—that is, one for which no data exists in the directory—by using `ldapmodify`. You can index an attribute for which data already exists in the directory by using the Catalog Management tool. You can drop an index from an attribute by using `ldapmodify`, but Oracle recommends that you use the Catalog Management tool.

#### Indexing an Attribute for Which No Data Exists by Using ldapmodify

Once you have defined a new attribute in the schema, you can add it to the catalog entry by using `ldapmodify`. To add an attribute for which no directory data exists by using `ldapmodify`, import an LDIF file by using `ldapmodify`. For example, to add a new attribute `foo` that has already been defined in the schema, import the following LDIF file by using `ldapmodify`:

```ldif
dn: cn=catalogs
changetype: modify
add: orclindexedattribute
orclindexedattribute: foo
```

You should not use this method to index an attribute for which data exists in the directory. To index such an attribute, use the Catalog Management tool.

#### Dropping an Index from an Attribute by Using ldapmodify

To drop an index from an attribute by using `ldapmodify`, specify `delete` in the LDIF file. For example:

```ldif
dn: cn=catalogs
changetype: modify
delete: orclindexedattribute
orclindexedattribute: foo
```

See Also: The `ldapmodify` command-line tool reference in *Oracle Identity Management User Reference*
How to Extend the Number of Attributes Associated with Entries

Indexing an Attribute for Which Data Exists by Using the Catalog Management Tool

Use the Catalog Management tool to index an attribute for which data already exists and to drop an index from an attribute.

See Also: The catalog command-line tool reference in Oracle Identity Management User Reference

Note: Unless you are absolutely sure that the indexes were not created by the base schema that was installed with Oracle Internet Directory, be careful not to use the catalog delete=T option to remove indexes from attributes. Removing indexes from base schema attributes can adversely impact the operation of Oracle Internet Directory.

How to Extend the Number of Attributes Associated with Entries

You can extend the number of attributes for entries. The method you use depends on whether the entries already exist.

For an existing entry, there are two ways to extend the attributes associated with it. One way is to add names of object classes to the list in the objectclass attribute for each entry. If your directory is relatively small, then this can be a desirable method because it enables searches for entries based on that attribute. However, if your directory is large, then entering the names of object classes to the objectclass attribute can be very painstaking. In this case, the second way, namely, using content rules, may be a more efficient way to extend the content of entries.

This section contains these topics:
- Extending the Number of Attributes Prior to Creating Entries in the Directory
- Extending the Number of Attributes for Existing Entries by Creating an Auxiliary Object Class
- Extending the Number of Attributes for Existing Entries by Creating a Content Rule

Extending the Number of Attributes Prior to Creating Entries in the Directory

At installation, Oracle Internet Directory provides standard LDAP object classes and several proprietary object classes. You cannot add mandatory attributes to the sets of attributes belonging to these predefined object classes. If a given object class does not contain all the attributes that you want for an entry, then you can do one of the following:

- Define a new (base) object class
- Define an object subclass

See Also:
- "Oracle Identity Management LDAP Object Classes" in Oracle Identity Management User Reference for a list of object classes in the schema installed with Oracle Internet Directory
- About Object Class Management on page 11-2 for instructions on how to define a new object class or object subclass
Extending the Number of Attributes for Existing Entries by Creating an Auxiliary Object Class

You can create an auxiliary object class containing the additional attributes you want for your entry, and then associate that auxiliary object class with the entry. You associate the auxiliary object class with the entry by specifying it in the objectClass attribute for the entry.

See Also:
- "About Object Class Management" on page 11-2 for instructions on creating auxiliary object classes
- Chapter 8, "Directory Entries Administration" for instructions on associating an object class with an entry

Extending the Number of Attributes for Existing Entries by Creating a Content Rule

A content rule, following your specifications, determines the kind of content allowed in any entry that is associated with a particular structural object class. For example, you can specify that any entry associated with the person object class must have, in addition to the attributes in that object class, other attributes as well. The additional attributes can be those of an auxiliary object class, and they can be either mandatory or optional.

Whereas you must list auxiliary classes in the entry—which can be an administrative burden—you do not need to list content rules in the entry.

In addition to the structural object class to which it applies, a content rule can also indicate:
- Auxiliary object classes allowed for entries governed by the rule
- Mandatory attributes, in addition to those called for by the structural and auxiliary object classes, required for entries governed by the DIT content rule
- Optional attributes permitted for entries governed by the DIT content rule, in addition to those called for by structural and auxiliary object classes

Rules for Creating and Modifying Content Rules

Content rules are defined as values of the DITContentRule attribute in the subschema subentry (cn=subschemasubentry). They must conform to these rules:
- The structural object class of the entry identifies the content rule applicable for the entry. If no content rule is present for a structural object class, then entries associated with that object class contain only the attributes permitted by the structural object class definition.
- Because a content rule is associated with a structural object class, all entries of the same structural object class have the same content rule regardless of the their location in the DIT.
- The content of an entry must be consistent with the object classes listed in the objectClass attribute of that entry. More specifically:
  - Mandatory attributes of object classes listed in the objectClass attribute must always be present in the entry.
  - Optional attributes of auxiliary object classes indicated by the content rule can also be present even if the objectClass attribute does not list these auxiliary object classes.
Schema Enforcement When Using Content Rules

When validating an object for schema consistency, the directory server uses the content rule for the structural object class of the entry. It also uses all the other object classes listed in the entry.

If more than one content rule exists for an object class, then, when adding or modifying an entry, or when bulkloading data, the following rules apply.

- An entry can have attributes from all the auxiliary object classes listed in the various content rules. Not specifying an object class in the content rule does not restrict a client from explicitly adding an auxiliary object class in directory entries.

- An entry must contain values for all the mandatory attributes listed in:
  - The content rules
  - The object classes associated with the entry
  - The auxiliary object classes listed in the content rule applicable to the entry

- Optionally, an entry can contain values for any or all the optional attributes listed in:
  - The content rule
  - The object classes listed in the entry
  - The auxiliary object classes listed in the content rule applicable for the entry

- If any attribute is specified as mandatory, then it overrides any other definition that defines it as optional.

Searches for Object Classes Listed in Content Rules

Because the auxiliary object classes listed in content rules are not listed in the objectclass attribute for an entry, you cannot list those object classes as filters when you search for entries. Instead, base your searches on the structural object class that you are interested in. If you need to base your search on an auxiliary object class, then add that auxiliary object class to the objectclass attribute in the user objects explicitly.

For example, a content rule for structural object class inetOrgPerson may specify an auxiliary object class orclUser. However, this does not mean that every inetOrgPerson entry in the directory contains orclUser as a value of the objectclass attribute. As a result, the search with the filter objectclass=orclUser fails. Instead of querying for an auxiliary object class contained in the content rule, you should query for structural object classes—for example, objectclass=inetOrgPerson.

To base a search on objectclass=orcluser, add orclUser as one of the values of objectclass attribute in each entry.

These considerations apply also to filters used in access control policies. If you are using a content rule to associate additional auxiliary object classes, then use only the structural object classes in the search filters.
Managing Content Rules
This section tells you how to manage content rules by using Oracle Directory Manager and command-line tools.

Managing Content Rules by Using Oracle Directory Manager
This section tells you how to use Oracle Directory Manager to create and modify content rules.

Creating a Content Rule by Using Oracle Directory Manager
To create a content rule:

1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance.
2. Select Schema Management.
3. In the right pane, select the Content Rules tab.
5. In the New Content Rule dialog box, enter values in the appropriate fields. These fields are described in Table A–38 on page A-22.
6. Choose OK.

Modifying a Content Rule by Using Oracle Directory Manager
To modify a content rule:

1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance.
2. Select Schema Management.
3. In the right pane, select the Content Rules tab.
4. Select the content rule you want to modify, then choose Edit. The Content Rule dialog box appears.
5. In the Content Rule dialog box, enter values in the appropriate fields. The fields for this dialog box are described in Table A–39 on page A-23.
6. Choose OK.

Managing Content Rules by Using Command-Line Tools
The format of a content rule is:

```
DITContentRule ::= SEQUENCE {
  oids                      ALPHA-NUMERIC-OID,
  structuralObjectClass     OBJECT-CLASS,
  LABEL                     CONTENT-LABEL OPTIONAL,
  auxiliaries               SET (1..MAX) OF OBJECT-CLASS OPTIONAL,
  mandatory                 SET (1..MAX) OF ATTRIBUTE OPTIONAL,
  optional                  SET (1..MAX) OF ATTRIBUTE OPTIONAL,
  precluded                 SET (1..MAX) OF ATTRIBUTE OPTIONAL
}
```

Table 11–1 describes the parameters. Note that the attribute and object class names are case-insensitive.
During the process of defining a new content rule, the directory server validates the syntax and ensures that the attributes and object classes listed in the content rule have been defined in the directory.

Content rules can be specified for structural object classes only. The name of the object class is case-insensitive.

You can specify more than one content rule for each structural object class provided the content rules have different labels associated with them.

To modify an existing definition of a content rule, the client must first delete the existing definition and then add the new definition. Simple replacement of a content rule by using the `replace` command is not allowed.

To delete a content rule, the client needs to specify only the structural object class and the alphanumeric object identifier of the content rule. Optionally, the client can also specify the associated version of the content rule to be deleted.

### Attribute Aliases In the Directory

This section includes the following topics:

- Features of Attribute Aliases
- Attribute Alias Rules
- Managing Attribute Aliases by Using Command-Line Tools
- Using Attribute Aliases
- Object Identifier Support in LDAP Operations

### Features of Attribute Aliases

As of 10g (10.1.4.0.1), you can create aliases for attribute names. For example, you could create the user-friendly alias `surname` for the attribute `sn`. Once you create an alias for an attribute name, a user can specify the alias instead of the attribute name in an LDAP operation.
You define an alias for an attribute in the LDAP schema definition of the attribute. The directory schema operational attribute `attributeTypes` has been enhanced to allow you to include aliases in the attribute name list. In previous releases, the format for an attribute name list was:

```plaintext
attributeTypes=( ObjectIdentifier NAME 'AttributeName' ... )
```

As of 10g (10.1.4.0.1), you may optionally specify:

```plaintext
attributeTypes=( ObjectIdentifier NAME ( 'AttributeName' 'Alias1' 'Alias2' ...) ... )
```

This is consistent with the LDAP protocol as specified by RFC 2251 and RFC 2252. In the attribute name list, the first item is recognized as the name of the attribute and rest of the items in the list are recognized as attribute aliases. For example, to specify the alias `surname` for the attribute `sn`, you would change the schema definition for `sn` from:

```plaintext
attributeTypes=( 2.5.4.4 NAME 'sn' SUP name )
```

to:

```plaintext
attributeTypes=( 2.5.4.4 NAME ( 'sn' 'surname' ) SUP name )
```

### Attribute Alias Rules

The following rules apply to attribute aliases:

- An attribute alias name must be unique throughout all the actual attribute names and other attribute aliases across all the schema components.
- When you define an attribute, the first value in the `attributeTypes` definition `NAME` field must be the actual attribute name. You define attribute aliases in the `NAME` field after the actual attribute name.
- Attribute alias names follow the same syntax rules as attribute names.
- You delete an attribute alias by redefining the attribute without the alias.

---

**Note:** DN is not an attribute. You cannot define `dn` in the schema. Therefore you cannot create an alias for `dn`.

---

### Managing Attribute Aliases by Using Command-Line Tools

As of 10g (10.1.4.0.1), you cannot use Oracle Directory Manager to manage attribute aliases. You add, modify, or delete attribute aliases by creating an LDIF file, then using `ldapmodify` with the following syntax:

```plaintext
ldapmodify -h host -p port -f ldif_file_name
```

The following examples show LDIF file formats you would use to perform these tasks:

- **Adding a New Attribute With Attribute Aliases**
- **Adding or Modifying Attribute Aliases in Existing Attributes**
- **Deleting Attribute Aliases**

#### Adding a New Attribute With Attribute Aliases

The following LDIF file adds the attribute `myattr` with the attribute aliases `myalias1` and `myalias2:`
Adding or Modifying Attribute Aliases in Existing Attributes

The following LDIF file adds the attribute aliases surname and mysurName to the existing attribute sn:

dn: cn=subschemasubentry
changetype: modify
delete: attributeTypes
attributeTypes: ( 2.5.4.4 NAME 'sn' SUP name )
-
add: attributeTypes
attributeTypes: ( 2.5.4.4 NAME ( 'sn' 'surname' 'mysurName' ) SUP name )

Deleting Attribute Aliases

The following LDIF file deletes the attribute alias mysurName but not the attribute alias surName from the attribute sn:

dn: cn=subschemasubentry
changetype: modify
delete: attributeTypes
attributeTypes: ( 2.5.4.4 NAME ( 'sn' 'surname' 'mysurName' ) SUP name )
-
add: attributeTypes
attributeTypes: ( 2.5.4.4 NAME ( 'sn' 'surname' ) SUP name )

The following LDIF file deletes both attribute aliases, surname and mysurName, from the attribute sn:

dn: cn=subschemasubentry
changetype: modify
delete: attributeTypes
attributeTypes: ( 2.5.4.4 NAME ( 'sn' 'surname' 'mysurName' ) SUP name )
-
add: attributeTypes
attributeTypes: ( 2.5.4.4 NAME 'sn' SUP name )

Using Attribute Aliases

Once you have defined attribute aliases in the LDAP schema, users can substitute the aliases for attribute names in LDAP operations. The following examples show the commands to use and the results to expect:

- Using Attribute Aliases with ldapsearch
- Using Attribute Aliases with ldapadd
- Using Attribute Aliases with ldapmodify
- Using Attribute Aliases with ldapdelete
- Using Attribute Aliases with ldapmoddn

Table 11–2 shows the aliases used in the examples and the attributes names they represent:
Using Attribute Aliases with ldapsearch

The LDAP server recognizes attribute aliases in the search filter string, in the base DN, and in the required attributes list in `ldapsearch` operations. The search result will contain the actual attribute names, unless the user explicitly asks for an alias using the required attributes list. For example, suppose the user specifies the following search, using the aliases `organizationalUnit`, `country`, and `organization` in the base DN and the alias `surname` in the filter string:

```
ldapsearch -p 389 -h myhost \
    -b "organizationalUnit=dev,country=us,organization=myorg" \
    -s sub "surname=brown"
```

The search will return a result similar to this:

```
uid=mbrown,ou=dev,c=us,o=myorg
uid=mbrown
sn=Brown
cn=Mark Brown
telephonenumber;office=444006
telephonenumber;mobile=555006
objectclass=organizationalPerson
objectclass=top
objectclass=person
```

Now suppose the user specifically asks for the aliases `surname`, `commonname`, and `userid` by including them in a required attributes list, like this:

```
ldapsearch -p 389 -h myhost \
    -b "organizationalUnit=dev,country=us,organization=myorg" \
    -s sub "surname=brown" surname commonname userid phone
```

Because the user specifically included the aliases, the search will return a result similar to this:

```
uid=mbrown,ou=dev,c=us,o=myorg
surname=Brown
commonname=Mark Brown
userid=mbrown
phone;office=444006
phone;mobile=555006
```

Table 11–2  Attribute Aliases Used in Examples

<table>
<thead>
<tr>
<th>Alias</th>
<th>Attribute Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>userid</td>
<td>uid</td>
</tr>
<tr>
<td>organizationalunit</td>
<td>ou</td>
</tr>
<tr>
<td>country</td>
<td>c</td>
</tr>
<tr>
<td>organization</td>
<td>o</td>
</tr>
<tr>
<td>surname</td>
<td>sn</td>
</tr>
<tr>
<td>commonname</td>
<td>cn</td>
</tr>
<tr>
<td>phone</td>
<td>telephonenumber</td>
</tr>
</tbody>
</table>
Using Attribute Aliases with ldapadd

The LDAP server recognizes attribute aliases in place of attribute names during the add operation. When the LDAP server stores the entry, it replaces the alias with the actual attribute name.

The command-line format is:

```
ldapadd -h host -p port -f ldif_file_name
```

The user could provide an LDIF file like this:

```
dn: userid=mbrown,organizationalUnit=dev,country=us,organization=myorg
objectclass: account
objectclass: organizationalPerson
userPassword: mbrown
surname: Brown
commonName: Mark Brown
userPassword: welcome
phone;office: 444006
phone;mobile: 555006
```

The entry will be stored as if the file contained the attribute names instead of the aliases. On subsequent LDAP searches, however, the DN will be returned as it was entered when added or modified:

```
dn: userid=mbrown,organizationalUnit=dev,country=us,organization=myorg
```

This is standard behavior for LDAP search results. The DN is always returned with the same format that was used when the entry was created.

Using Attribute Aliases with ldapmodify

The LDAP server recognizes attribute aliases in place of attribute names during the modify operation.

The command-line format is:

```
ldapmodify -h host -p port -f ldif_file_name
```

The user could provide an LDIF file like this:

```
dn: userid=mbrown,organizationalUnit=dev,country=us,organization=myorg
changeType: modify
replace: surname
surname: davis
```

The entry will be stored as if the file contained the attribute names instead of the aliases. On subsequent LDAP searches, however, the DN will be returned as it was entered when added or modified:

```
dn: userid=mbrown,organizationalUnit=dev,country=us,organization=myorg
```

This is standard behavior for LDAP search results. The DN is always returned with the same format that was used when the entry was created.

Using Attribute Aliases with ldapdelete

The LDAP server recognizes attribute aliases in the DN provided for delete operations. For example, suppose the user provides a request like this, with the aliases userid, organizationalUnit, country, and organization in the search filter:
Using Attribute Aliases with `ldapmoddn`

The LDAP server recognizes attribute aliases in the DN, the new RDN, and the new parent DN options. For example, suppose the user types the command line:

```
ldapmoddn -b "userid=mbrown,organizationalUnit=dev,country=us,organization=myorg" \\
-R "userid=mdavis"
```

The LDAP server interpret the command line as if the user had typed

```
ldapmoddn -b "uid=mbrown,ou=dev,c=us,o=myorg" \\
-R "uid=mdavis"
```

The entry will be stored as if the file contained the attribute names instead of the aliases. On subsequent LDAP searches, however, the DN will be returned as it was entered when added or modified:

```
dn: userid=mbrown,organizationalUnit=dev,country=us,organization=myorg
```

This is standard behavior for LDAP search results. The DN is always returned with the same format that was used when the entry was created.

Object Identifier Support in LDAP Operations

Users can substitute object identifiers for attribute names in the same way as attribute aliases.

Matching Rules in the Directory

This section contains these topics:

- Viewing Matching Rules by Using Oracle Directory Manager
- Viewing Matching Rules by Using `ldapsearch`

Note: Matching rules cannot be modified.

Viewing Matching Rules by Using Oracle Directory Manager

1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance.
2. Select Schema Management.
3. In the right pane, select the Matching Rules tab. The fields in this tab page are shown as column heads. They are described in Table A–37 on page A-22.

Viewing Matching Rules by Using `ldapsearch`

Use `ldapsearch` on the subentry `cn=subSchemaSubentry`. 
See Also: The ldapsearch command-line tool reference in Oracle Identity Management User Reference

Syntaxes in the Directory

This section contains these topics:

- Viewing Syntaxes by Using Oracle Directory Manager
- Viewing Syntaxes by Using by Using ldapsearch

---

Note: Syntaxes cannot be modified.

---

Viewing Syntaxes by Using Oracle Directory Manager

To view syntaxes by using Oracle Directory Manager:

1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance.
2. Select Schema Management.
3. In the right pane, select the Syntaxes tab. The fields in this tab page are shown as column heads. They are:
   - Description: Name of the attribute syntax
   - Object ID: Unique identifier of this syntax

---

Viewing Syntaxes by Using by Using ldapsearch

Use ldapsearch on the subentry cn=subSchemaSubentry.

See Also: The ldapsearch command-line tool reference in Oracle Identity Management User Reference
Referential Integrity is a new feature of Oracle Internet Directory. If Referential Integrity is enabled, whenever you update an entry in the directory, the server also updates other entries that refer to that entry. For example, if you remove a user's entry from the directory, and the user is a member of a group, the server also removes the user from the group. If Referential Integrity is not enabled, the user remains a member of the group until manually removed.

Referential Integrity takes effect in two situations:

- **Delete**–When an entry is deleted, all the DN attributes that refer to this entry DN are removed.
- **Modify**–When an entry's DN is modified (renamed), all the attributes that refer to this entry DN are modified.

This chapter contains the following topics:

- Configuring and Enabling Referential Integrity
- Disabling Referential Integrity

### Configuring and Enabling Referential Integrity

To configure and enable Referential Integrity, perform the following steps:

**1.** Edit the file `$ORACLE_HOME/ldap/server/plugin/rimoddn.java`, as follows:

   a. Go to line 107 in the file. It looks like this:

      ```java
      conn = DriverManager.getConnection("jdbc:oracle:thin:ods/ODSPWD@OHOST:OPORT:OSID");
      ```

   b. Replace `ODSPWD` with your Oracle Internet Directory ODS password.

   c. Replace `OHOST` with your Oracle Internet Directory back end database host name.

   d. Replace `OPORT` with your Oracle Internet Directory back end database port number.

   e. Replace `OSID` with your Oracle Internet Directory back end database `ORACLE_SID`.

**2.** Compile `$ORACLE_HOME/ldap/server/plugin/rimoddn.java`, as follows:

   ```bash
   % $ORACLE_HOME/jdk/bin/javac -classpath \n   $ORACLE_HOME/jdbc/lib/classes12.zip:$ORACLE_HOME/ldap/jlib/ospf.jar \n   rimoddn.java
   ```
3. The file rimoddn.java now contains the ODS password in clear text. Either remove the password from the file or modify the permissions on the file to prevent unauthorized access.

4. Register the plug-ins:

   % ldapadd -h hostname -p port -D cn=orcladmin -w orcladmin_pwd -v -f \n   $ORACLE_HOME/ldap/admin/oidriplg.dat

   From this point on, all the ldapmoddn and ldapdelete operations on DN reference attributes are recorded.

   See Also: Chapter 32, "Oracle Internet Directory Server Plug-in Framework"

5. Edit $ORACLE_HOME/ldap/admin/oidrimdx.pls, as follows:

   a. Go to line 42 in the file. It looks like this:

      v_attrlist := ODS.MODDELREF.t_attrlist('uniquemember', 'owner');

      This line specifies the default values uniquemember and owner.

   b. If you have more DN attribute references to be modified, add the additional attributes to the line. For example, to specify that the DN attribute, manager be modified, change the line to:

      v_attrlist := ODS.MODDELREF.t_attrlist('uniquemember', 'owner', 'manager');

   Note:

   - Use all lowercase letters when specifying the attributes.
   - Each attribute you add must be searchable. If necessary, run the catalog command to index the attribute.

   See Also:

   - "catalog" on page 9-9.
   - "Indexing an Attribute by Using Command-Line Tools" on page 11-15

6. Run $ORACLE_HOME/ldap/admin/oidrimdx.pls at frequent intervals. The exact frequency depends upon site-specific needs, but every 15 minutes to 24 hours is a reasonable range.

   The script $ORACLE_HOME/ldap/admin/oidrimdx.pls consumes the records from the Referential Integrity storage table and deletes or modifies the DN references in all associated tables.

   The command to run it is:

   % sqlplus ods/odspassword@connect_string @$ORACLE_HOME/ldap/admin/oidrimdx.pls

   To run this command, you must be either the dba or Oracle Internet Directory administrator. On a UNIX or Linux system, you can set up a cron job to run the program as one of those users. If you incorporate the command line into a script or
Disabling Referential Integrity

7. If the entry cache is enabled, invalidate it every time you run $ORACLE_HOME/ldap/admin/oidrimdx.pls. The oidrimdx.pls program modifies tables in the database directly, which makes entries in the cache incorrect. Each time you run oidrimdx.pls, quickly disable and then enable the entry cache, as follows:

% ldapmodify -h host -p port -D cn=orcladmin -w orcladmin_pwd <<EOF
dn:
  changetype: modify
  replace: orclecacheenabled
  orclecacheenabled: 0
EOF

% ldapmodify -h host -p port -D cn=orcladmin -w orcladmin_pwd <<EOF
dn:
  changetype: modify
  replace: orclecacheenabled
  orclecacheenabled: 1
EOF

If you are running oidrimdx.pls from a cron job, and the entry cache is enabled, you should include the commands to invalidate the cache.

Disabling Referential Integrity

To disable Referential Integrity, perform the following steps:

1. Delete the plug-ins, as follows:

% ldapdelete -h hostname -p port -D cn=orcladmin -w orcladmin_pwd "cn=ri_postdelete,cn=plugin,cn=subconfigsubentry"
% ldapdelete -h hostname -p port -D cn=orcladmin -w orcladmin_pwd "cn=ri_postmoddn,cn=plugin,cn=subconfigsubentry"

2. Stop running $ORACLE_HOME/ldap/admin/oidrimdx.pls. If you have been using a cron job, delete or disable it.
This chapter explains how to administer both static and dynamic groups in Oracle Internet Directory. This chapter contains these topics:

- **About Groups**
- **Managing Group Entries**

### About Groups

Oracle Internet Directory enables you to assign and manage membership in two types of groups—namely, static groups and dynamic groups. Each type of group suited for a different purpose.

This section contains these topics:

- **Static Groups**
- **Dynamic Groups**
- **Hierarchies**
- **Querying Group Entries**
- **When to Use Each Kind of Group**

### Static Groups

A static group is one whose entry contains a list of members that you explicitly administer.

A static group requires you to explicitly administer its membership. For example, if a member changes his name, then you need to change that user’s DN for each group he belongs to. For this reason, a static group is best suited for a group whose membership is unlikely to change frequently. Moreover, because a static group contains a list of member DNs, its footprint in the directory increases with the membership list. For this reason, it is best suited for a group whose entries take up relatively less space in the directory.

**Schema Elements for Creating Static Groups**

When you create the entry for this kind of group, you associate it with either the `groupOfNames` or `groupOfUniqueNames` object class.

Each of these object classes has a multivalued attribute for storing the names of group members. To assign a user as a member of a group, you add the DN of each member to
the respective multivalued attribute. Conversely, to remove a member from a group, you delete the member's DN from the respective attribute. In the groupOfNames object class, this multivalued attribute is member, and, in the groupOfUniqueNames object class, it is uniqueMember.

Dynamic Groups

A dynamic group is one whose membership, rather than being maintained in a list, is computed, based on rules and assertions you specify. As of Oracle Internet Directory 10g (10.1.4.0.1), dynamic groups based on labeleduri attributes are cached.

By cached, we mean that dynamic group members are computed when the dynamic group is added, and that the member list is kept consistent when the dynamic group is later modified. As entries are added, modified, deleted, and renamed, the member lists of all dynamic groups are kept consistent. For example, if there is a dynamic group containing all person entries under "c=us", when we add "cn=user1,c=us", that entry is automatically added to the member list of the dynamic group. Similarly, when we delete "cn=user1,c=us", the entry is removed from the dynamic group's member list. This feature ensures that whenever a search is performed for a dynamic group, the member list can be returned without any additional computation. The search performance for dynamic groups is now the same as for static groups.

Dynamic groups can have static as well as dynamic members. The static members are listed as values of the member or uniquemember attribute.

---

**Note:** Only dynamic groups based on labeleduri attributes are cached. Dynamic groups based on CONNECT_BY assertion are not cached.

---

**Note:** You cannot add a dynamic group based on the labeledURI attribute with scope base. Only scope sub and one are supported.

---

**Note:** To refresh dynamic group memberships, set the attribute orclrefreshdgrmems in the DSA Configuration entry to 1. Oracle Internet Directory will recompute the member lists for all dynamic groups and reset the value of orclrefreshdgrmems to 0.

---

**Note:** As of 10g (10.1.4.0.1), when you query for the groups that a user belongs to, dynamic groups are automatically included in the result.

In earlier releases, you had to pass a control to direct that dynamic groups, in addition to static groups, be queried. In releases prior to 10g (10.1.4.0.1), if this control was not passed, then only static groups were queried.

For more information on controls used by Oracle Internet Directory, see "About LDAP Controls" in Oracle Identity Management User Reference.
Enhancements to and Limitations of Dynamic Groups in Oracle Internet Directory 10g (10.1.4.0.1)

As of Oracle Internet Directory 10g (10.1.4.0.1), you can use dynamic groups in the same ways you use static groups. For example, you can use them in:

- Access control lists, by associating the group with either the orclACPgroup or the orclPrivilegeGroup object class.
- Searches for required attributes of members. For dynamic groups, you would use the -G or the -C control to do this, and for static groups you would use the -C control.
- Hierarchical group resolution queries

Dynamic groups have the following limitations in Oracle Internet Directory 10g (10.1.4.0.1):

- Only dynamic groups based on labeleduri attributes are cached. Dynamic groups based on CONNECT_BY assertions are not cached.
- Hierarchical queries and queries involving specific attributes of members can only be done on cached dynamic groups.
- Dynamic groups can only be added using ldapadd. They cannot be added by using bulkload.
- If the catalog tool is used to drop and re-create the ct_member or ct_uniqueMember catalog tables, the dynamic group member lists must be recomputed by setting the orclrefreshdgrmems attribute of the DSA Configuration entry to 1 using ldapmodify.

Schema Elements for Creating a Dynamic Group

When you create a dynamic group, you begin as when creating a static group—that is, you associate its entry with either the groupOfNames or groupOfUniqueNames object class. You then associate that object class with the auxiliary object class orclDynamicGroup. This auxiliary object class has various attributes in which you specify one of two methods for dynamically computing the membership of the group.

The two methods are:

- Using the labeledURI attribute
  
  When using this method, the directory server performs a typical search based on the hierarchy of the DIT. It requires you to provide a value for one of the attributes of the orclDynamicGroup object class, namely labeledURI. In this attribute, you specify the base of the query, the filters, and any required attributes. For example, suppose that you have entered the following value for the labeledURI attribute:

  labeledURI:ldap://host:port/ou=NewUnit,o=MyCompany,c=US??sub?(objectclass=person)
When you use this method, a search for the entry returns entries for all members of the group.

Do not set `orclConnectByAttribute` or `orclConnectByStartingValue` when using the `labeledURI` attribute method.

**See Also:** "The LDAP URL Format" (RFC 2255). T. Howes, M. Smith, December 1997. This RFC provides more information about how LDAP URLs are to be represented—as, for example, in the `labeledURI` attribute. It is available on the World Wide Web at [http://www.ietf.org](http://www.ietf.org).

- Using a `CONNECT BY` assertion

Unlike the previous method, this method relies not on the hierarchy of the DIT, but on attributes that implicitly connect entries to each other, regardless of their location in the DIT. For example, the `manager` attribute connects the entries of employees with those of their managers, and this connection applies regardless of the location of the employee entries in the DIT. This method uses a `CONNECT BY` clause in which you specify the attribute to use for building the hierarchy—for example, `manager`—and the starting value for such a hierarchy—for example, `cn=Anne Smith`.

**See Also:** Performing Hierarchical Searches in *Oracle Identity Management Application Developer’s Guide*

More specifically, to use this method, you specify in the `orclDynamicGroup` object class a value for each of the single-valued attributes in Table 13–1.

**Table 13–1 orclDynamicGroup Attributes for "Connect By" Assertions**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>orclConnectByAttribute</code></td>
<td>The attribute that you want to use as the filter for the query—for example, <code>manager</code></td>
</tr>
<tr>
<td><code>orclConnectByStartingValue</code></td>
<td>The DN of the attribute you specified in the <code>orclConnectByAttribute</code> attribute—for example, <code>cn=Anne Smith</code></td>
</tr>
</tbody>
</table>

Do not set `labeledURI` when using the `CONNECT BY` assertion method.

For example, to retrieve the entries of all employees who report to Anne Smith in the MyOrganizational Unit in the Americas, you would provide values for these attributes as follows:

```
orclConnectByAttribute=manager
orclConnectByStartingValue=
"cn=Anne Smith,ou=MyOrganizationalUnit,o=MyCompany,c=US"
```

You can also develop an application specifying that you want the values for a particular attribute—for example, the `email` attribute—of all the members.

**See Also:** *Oracle Identity Management Application Developer’s Guide* for more information about how to develop applications that retrieve values for particular attributes.

**Hierarchies**

Hierarchies can be either explicit or implicit.
In explicit hierarchies, the relationship is determined by the location of the entry in the DIT—for example, Group A may reside higher in the DIT than Group B.

In implicit hierarchies, the relationship between entries is determined not by the location in the DIT, but by the values of certain attributes. For example, suppose that you have a DIT in which the entry for John Doe is at the same level of the hierarchy as Anne Smith. However, suppose that, in the entry for John Doe, the manager attribute specifies Anne Smith as his manager. In this case, although their locations in the DIT are at an equal level, their rankings in the hierarchy are unequal because Anne Smith is specified as John Doe's manager.

---

**Note:** If you create a hierarchical group, be sure that it is truly hierarchical. For example, in a true hierarchy, Group A can be a member of Group B, but Group B cannot at the same time be a member of Group A. Because the latter relationship is cyclical, a search for the members of Group A fails.

In a query based on an implicit hierarchy, the client can specify in the search request the control 2.16.840.1.113894.1.8.3. The filter in this query specifies the attribute used to build the implicit hierarchy. For example, \((\text{manager=cn=john\ doe, o=foo})\) specifies the query for all people reporting directly or indirectly to John Doe. The implicit hierarchy is based on the manager attribute. The base of the search is ignored for such queries.

For more information on controls used by Oracle Internet Directory, see "About LDAP Controls" in *Oracle Identity Management User Reference*.

---

**See Also:** The C API chapter in *Oracle Identity Management Application Developer’s Guide*

---

**Querying Group Entries**

An application can query either kind of group to do the following:

- List all members of a group
- List all groups of which a user is a member
- Check to see if a user is a member of a particular group

In addition, you can query dynamic groups, but not static ones, for whatever member attributes you specify.

---

**When to Use Each Kind of Group**

When deliberating about which kind of group to use, you need to weigh the ease of administration against higher performance. For example, dynamic groups provide for easier administration, but cause a decrease in performance. Table 13–2 lists some things to consider when deliberating whether to use static or dynamic groups.
Managing Group Entries

This section contains these topics:

- Managing Static Group Entries by Using Oracle Directory Manager
- Managing Static Group Entries by Using Command-Line Tools
- Examples of Dynamic Group Entries
- Managing Dynamic Groups by Using Oracle Directory Manager
- Managing Dynamic Groups by Using Command-Line Tools

Table 13–2  Static and Dynamic Group Considerations

<table>
<thead>
<tr>
<th>Consideration</th>
<th>Static Groups</th>
<th>Dynamic Groups</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ease of administration</td>
<td>More difficult to administer if group memberships are large and change frequently</td>
<td>Easier to use, especially when group memberships are large and change frequently</td>
</tr>
<tr>
<td>Performance</td>
<td>Higher level of performance because you explicitly administer the membership list</td>
<td>Decreased level of performance because memberships are computed on the fly</td>
</tr>
<tr>
<td>Size of footprint in the directory</td>
<td>Larger footprint depending on the size of group memberships</td>
<td>Small footprint regardless of size of group memberships</td>
</tr>
</tbody>
</table>

Note: If you are creating a hierarchy of groups, be sure that it is a true hierarchy as described in "Hierarchies" on page 13-4.

See Also:

- "Security Groups" on page 18-3 for instructions on setting access control policies for group entries
- Globalization Support on page 3-15 and Chapter 18, "Directory Access Control" for information about access privileges

Managing Static Group Entries by Using Oracle Directory Manager

You can use Oracle Directory Manager to both create and modify static group entries.

Creating Static Group Entries by Using Oracle Directory Manager

If the entry belongs to the groupOfNames object class, then you determine membership in the group by adding DNs to the multivalued attribute member. If the entry belongs to the groupOfUniqueNames object class, then you determine membership in the group by adding DNs to the multivalued attribute uniqueMember.

To add a static group entry:

1. Expand in succession Oracle Internet Directory Servers and directory server instance.
2. Select Entry Management.
3. On the toolbar, choose Create. The New Entry dialog box appears.
4. In the **Distinguished Name** field, type the full DN. You may also use **Browse** to locate the DN of the parent for the entry you want to add, then type the RDN for the new entry, followed by a comma, to the left of that parent DN.

5. To specify the object classes you want to use for the new entry, to the right of the **Object Classes** box, choose **Add**. The Super Class Selector dialog box appears.

   a. In the Super Class Selector dialog box, select the following object classes:

      * top
      * Either **groupOfNames** or **groupOfUniqueNames**

   b. Choose **Select**. The object classes you selected appear in the **Object Classes** window of the New Entry dialog box.

6. Enter the mandatory and optional attributes for your group entry.

   If you selected the **groupOfNames** object class, a **Browse** button appears next to some of the fields, for example, the member field on the **Mandatory Properties** tab page. To enter a mandatory property by browsing:


   b. Use this dialog box to search for a particular entry you want to add to the list.

   c. In the **Distinguished Name** window of the Directory: Entry Management dialog box, select the entry, then choose **OK**. This returns you to the New Entry dialog box. The entry you just selected is added to the list in the members window.

7. Choose **OK**.

**Modifying a Static Group Entry by Using Oracle Directory Manager**

To modify the member list for a group entry:

1. Perform a search for the group entry you want to modify.

2. In the right pane, in the **Distinguished Name** box, select the group entry you want to modify.

3. Choose **Edit**.

4. In the Entry dialog box, scroll to the text area for the **member** attribute and modify the value.

5. Choose **OK**.

**Managing Static Group Entries by Using Command-Line Tools**

This section provides examples of how you create and modify static group entries.

**Creating a Static Group Entry by Using ldapadd**

The syntax for the LDIF file is:

```

dn: DN_of_group_entry
objectclass: top
objectclass: [groupOfNames] [groupOfUniqueNames]
member: DN of member 1
member: DN of member 2
```

The following command adds this LDIF file to the directory:

```
ldapadd -p port_number -h host -f file_name.ldif
```

**Example: Creating a Static Group Entry by Using ldapadd**
The following example shows an LDIF file named `myStaticGroup.ldif` for the entry for a group named `MyStaticGroup`:

```
dn: cn=myStaticGroup,c=us
objectclass: top
objectclass: groupOfNames
member: cn=John Doe
member: cn=Anne Smith
```

The following command adds this LDIF file to the directory:

```
ldapadd -p 389 -h myhost -f myStaticGroup.ldif
```

**Modifying a Static Group by Using ldapmodify**
To add a member to a group, the syntax of the LDIF file is:

```
dn: DN_of_group_entry
changetype: modify
add:member
member:DN_of_member_entry
```

To delete a member from a group, the syntax of the LDIF file is:

```
dn: DN_of_group_entry
changetype: modify
delete:member
member:DN_of_member_entry
```

Issue this command to modify the file:

```
ldapmodify -p 389 -v -f file_name.ldif
```

where `-v` specifies verbose mode.

**Example: Modifying a Static Group by Using ldapmodify**
The following example adds John Doe to a group named `MyStaticGroup`. As in the previous example, the data for this user entry is in the `myStaticGroup.ldif` file. This file contains the following:

```
dn: cn=myStaticGroup,c=us
changetype: modify
add:member
member: cn=John Doe
```

Issue this command to modify the file:

```
ldapmodify -p 389 -v -f myStaticGroup.ldif
```

where `-v` specifies verbose mode.
Examples of Dynamic Group Entries

This section provides examples of the two kinds of dynamic group entries.

Example: a Dynamic Group Entry Using the labeledURI Attribute
The following is an example of a dynamic group entry using the labeledURI attribute.

dn: cn=dgroup1
  cn: dgroup1
description: this is an example of a dynamic group
  labeleduri:ldap://hostname:7777/ou=oid,l=amer,dc=oracle,dc=dgrptest??sub?objectclass=person
objectclass: orcldynamicgroup
objectclass: groupOfUniqueNames
objectclass: top

This group will have uniquemember values that are the DNs of all entries associated with the object class person in the subtree ou=oid,l=amer,dc=oracle,dc=dgrptest.

Example: a Dynamic Group Entry Using the CONNECT BY Assertion
The following is an example of a dynamic group entry that uses the CONNECTBY assertion.

dn: cn=dgroup2
  cn: dgroup21
description: this is connect by manager assertion dynamic group
  orclconnectbyassertionbase: l=amer,dc=oracle,dc=dgrptest
  orclconnectbyattribute: manager
  orclconnectbystartingvalue: cn=john doe sr.
objectclass: orcldynamicgroup
objectclass: groupOfUniqueNames
objectclass: top

This dynamic group has unique members with values that are DNs of all the entries whose manager attribute is cn=john doe sr. either indirectly or directly. If several individuals have cn=john doe JR as their manager, and he, in turn, has cn=john doe SR. as his manager, then all the lower-level individuals are returned.

Managing Dynamic Groups by Using Oracle Directory Manager

You can use Oracle Directory Manager to both create and modify static group entries.

Creating Dynamic Group Entries by Using Oracle Directory Manager
If the entry belongs to the groupOfNames object class, then you determine membership in the group by adding DNs to the multivalued attribute member. If the entry belongs to the groupOfUniqueNames object class, then you determine membership in the group by adding DNs to the multivalued attribute uniqueMember.
To add a dynamic group entry:

1. Expand **Oracle Internet Directory Servers**, then **directory server instance**.
2. Select **Entry Management**.
3. On the toolbar, choose **Create**. The New Entry dialog box appears.
4. In the **Distinguished Name** field, type the full DN. You may also use **Browse** to locate the DN of the parent for the entry you want to add, then type the RDN for the new entry, followed by a comma, to the left of that parent DN.
5. To specify the object classes you want to use for the new entry, to the right of the **Object Classes** box, choose **Add**. The Super Class Selector dialog box appears.
   a. In the Super Class Selector dialog box, select the following object classes:
      * top
      * orcldynamicgroup
      * Either groupOfNames or groupOfUniqueNames
   b. Choose **Select**. The object classes you selected appear in the **Object Classes** window of the New Entry dialog box.
6. Enter the mandatory and optional attributes for your group entries.
   If you are using the labeledURI method for dynamically computing membership in the group, you must set the labeledURI attribute, but not the orclConnectByAttribute and orclConnectByStartingValue attributes. In the **Optional Properties** tab page, in the labeledURI field, specify the following:
   ```
   ldap:ldap_URL
   ```
   For example:
   ```
   ldap://my_host/ou=MyNeworganizationalUnit,
   o=MyCompany,c=US?sub?(objectclass=person)
   ```
   If you are using the CONNECT BY method for dynamically computing membership in the group, you must set the orclConnectByAttribute and orclConnectByStartingValue attributes, but not the labeledURI attribute. In the orclConnectByAttribute field, specify the attribute that you want to use as the filter for the query—for example, manager. In the orclConnectByStartingValue field, specify the DN of the attribute you specified in the orclConnectByAttribute attribute—for example, `cn=Anne Smith`.
   For information about specifying the other attributes that appear in the **Optional Properties** tab page, see "User and Group Schema Elements" in *Oracle Identity Management User Reference*.
   If you selected the groupOfNames object class, a **Browse** button appears next to some of the fields, for example, the member field on the **Mandatory Properties** tab page. If you choose Browse, the Directory: Entry Management dialog box appears. Use this dialog box to search for a particular entry you want to add to the list. Then, in the **Distinguished Name** window of the Directory: Entry Management dialog box, select the entry and choose **OK**. This returns you to the New Entry dialog box. The entry you just selected is added to the list in the members window.

7. Choose **OK**.
Modifying a Dynamic Group Entry by Using Oracle Directory Manager

To modify the member list for a dynamic group entry:

1. Perform a search for the group entry you want to modify.
2. In the right pane, in the **Distinguished Name** box, select the group entry you want to modify.
3. Choose **Edit**.
4. In the Entry dialog box, scroll to the text area for the *member* attribute and modify the value.
5. Choose **OK**.

Managing Dynamic Groups by Using Command-Line Tools

This section tells you how to create and modify dynamic groups by using command-line tools.

Creating a Dynamic Group Entry by Using ldapadd

If you use the *labeledURI* attribute, then the syntax for the LDIF file is:

```
 dn: DN_of_group_entry
 objectclass: top
 objectclass: [groupOfNames] [groupOfUniqueNames]
 objectclass: orclDynamicGroup
 labeledURI: ldap://ldap_URL
 member: DN of member 1
 member: DN of member 2
 .
 .
 member: DN of member N
```

The following command adds this LDIF file to the directory:

```
ldapadd -p port_number -h host -f file_name.ldif
```

If you use the *CONNECT BY* string, then the syntax for the LDIF file is:

```
 dn: DN_of_group_entry
 objectclass: top
 objectclass: [groupOfNames] [groupOfUniqueNames]
 objectclass: orclDynamicGroup
 orclConnectByAttribute: attribute_name
 orclConnectByStartingValue: DN_of_attribute
 member: DN of member 1
```

When specifying entries in this syntax, do not use double quotes around distinguished names.

Example: Creating a Dynamic Group Entry by Using ldapadd

The following example shows an LDIF file for the entry for a dynamic group:

```
 dn: cn=myDynamicGroup,c=us
 objectclass: top
 objectclass: groupOfNames
 objectclass: orclDynamicGroup
 labeledURI: ldap://my_host/ou=MyNeworganizationalUnit,
 o=MyCompany,c=US??sub?(objectclass=person)
```
member: cn=John Doe
member: cn=Anne Smith

The following command adds this LDIF file to the directory:

```
ldapadd -p 389 -h myhost -f myDynamicGroup.ldif
```

**Example: Modifying a Dynamic Group by Using ldapmodify**

To change the organizational unit of the group created in the previous example, the syntax of the LDIF file is:

```
dn: DN_of_group_entry
changetype: modify
replace:labeledURI
labeledURI:ldap://my_host/
ou=MyNeworganizationalUnit,o=MyCompany,c=US??sub?(objectclass=person)
```

**Note:** When you add or modify an entry, the Oracle directory server does not verify the syntax of the attribute values in the entry.
Logging, Auditing, and Monitoring the Directory

Oracle Internet Directory provides a comprehensive framework for enabling you to debug, audit, and monitor the directory. This chapter contains these topics:

- Log File Locations
- Using Debug Logging
- Using the Audit Log
- Monitoring Oracle Internet Directory Servers

Log File Locations

Oracle Internet Directory components output their log and trace information to log files in the ORACLE_HOME environment. Table 14–1 lists each component and the location of its corresponding log file.

<table>
<thead>
<tr>
<th>Component</th>
<th>Log File Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bulk Loader (bulkload)</td>
<td>$ORACLE_HOME/ldap/log/bulkload.log</td>
</tr>
<tr>
<td>Bulk Modifier (bulkmodify)</td>
<td>$ORACLE_HOME/ldap/log/bulkmodify.log</td>
</tr>
<tr>
<td>Bulk Delete Tool (bulkdelete)</td>
<td>$ORACLE_HOME/ldap/log/bulkdelete.log</td>
</tr>
<tr>
<td>Catalog Management Tool (catalog)</td>
<td>$ORACLE_HOME/ldap/log/catalog.log</td>
</tr>
<tr>
<td>Data Export Tool (ldifwrite)</td>
<td>$ORACLE_HOME/ldap/log/ldifwrite.log</td>
</tr>
<tr>
<td>Directory integration agent</td>
<td>$ORACLE_HOME/ldap/odi/log/AgentName.err where AgentName is the name of the agent</td>
</tr>
<tr>
<td>Directory integration server (odisrv)</td>
<td>$ORACLE_HOME/ldap/log/odisrvXX.log where XX is Oracle directory integration and provisioning server instance number</td>
</tr>
<tr>
<td>Directory replication server (oidrepld)</td>
<td>$ORACLE_HOME/ldap/log/oidrepld00.log</td>
</tr>
</tbody>
</table>
Using Debug Logging

This section contains these topics:

- About Oracle Internet Directory Debug Logging
- About Log Messages
- Setting Debug Logging Levels
- Setting the Operation Debug Dimension
- Force Flushing the Trace Information to a Log File

About Oracle Internet Directory Debug Logging

Oracle Internet Directory enables you to:

- View logging information for the directory server, the directory replication server, and the directory integration server
- Set the logging level
- Specify one or more operations for which you want logging to occur
- Search messages in a standard format to determine remedial action for fatal and serious errors
- View trace messages according to their severity and order of importance
- Diagnose Oracle Internet Directory components by examining trace messages with relevant information about, for example, entry DN, ACP evaluation, and the context of an operation

About Log Messages

This section discusses log messages—those associated with specified LDAP operations and those not. It provides an example of a trace log and explains how to interpret it.
Log Messages for Specified LDAP Operations

Log messages for a specified operation are stored as a trace object. This object tracks the operation from start to finish across the various Oracle Internet Directory modules. It is entered in the log file when one of the following occur:

- An LDAP operation completes
- A high priority message is logged
- The trace messages buffer is full

Each thread has one contiguous block of information for each operation, and that block is clearly delimited. This makes it easy, in a shared server environment, to follow the messages of different threads, operations, and connections.

If, because of an internal message buffer overflow, a single trace object cannot contain all the information about an operation, then the information is distributed among multiple trace objects. Each distributed piece of information is clearly delimited and has a common header. To track the progress of the operation, you follow the trace objects and their common header to the end, which is marked with the trace message "Operation Complete".

Log Messages Not Associated with Specified LDAP Operations

Messages not associated with any LDAP operation are represented in a simple format, which is not object-based. It is entered in the log file when either the operation completes or a high priority message is encountered.

Example: Trace Messages in Oracle Internet Directory Server Log File

```
2003/01/28:13:44:27 * Main:1 * Starting up the OiD Server, on node dthakuri-sun


2003/01/28:13:44:31 * ServerWorker:6 * INFO * gslsfAStr2Filter * Filter="(|(objectclass=referral))"
2003/01/28:13:44:31 * ServerWorker:6 * INFO * gslsfAStr2Filter * Filter="(objectclass=referral)"
2003/01/28:13:44:31 * ServerWorker:6 * INFO * gslsfCStr2Simple * Filter="objectclass=referral"
2003/01/28:13:44:31 * ServerWorker:6 * INFO * gslsbnrNormalizeString() String to Normalize: "objectclass"
2003/01/28:13:44:31 * ServerWorker:6 * INFO * gslsbnrNormalizeString() Normalized value: "objectclass"

BEGIN
2003/01/28:13:45:49 * ServerWorker:6 * ConnID:0 * OpId:0 * OpName:bind
13:45:49 * INFO * gslfbiAddBind * Entry
13:45:49 * INFO * gslfbiGetControlInfo * Entry
```
How to Interpret Trace Messages in the Log File

As shown in the sample messages in the previous section, log information can be associated with either a thread that performs an operation or one that does not. In the case of a thread that performs an operation, the header of the log contains:

- Date and time
- Thread name and identifier for the particular connection
- Connection identifier
The name and identifier of the associated operation

A thread that does not perform an operation logs normal trace messages. Its header contains the date, time, and the thread identifier. It does not contain connection and operation-related information.

A trace object starts with the keyword BEGIN and ends with the keyword END.

Table 14–2 describes each field in a trace message.

<table>
<thead>
<tr>
<th>Field 1</th>
<th>Field 2</th>
<th>Field 3</th>
<th>Field 4</th>
<th>Field 5</th>
<th>Field 6</th>
</tr>
</thead>
<tbody>
<tr>
<td>For messages not based on objects:</td>
<td>For non-object-based trace messages only, the thread identifier</td>
<td>Trace message criticality. This has four possible values:</td>
<td>Function name</td>
<td>Information about the operation performed. This information can be used to diagnose problems.</td>
<td>Error code, if available. The error code could be for the operating system, the Oracle database, or LDAP.</td>
</tr>
<tr>
<td>Date and time</td>
<td>Time only</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Setting Debug Logging Levels

You can set debug logging levels by using either Oracle Directory Manager or the OID Control Utility.

Setting Debug Logging Levels by Using Oracle Directory Manager

To set the debug logging level:

1. In the Navigator pane, expand Oracle Internet Directory Servers and select a server instance. The group of tab pages for that server appear in the right pane.
2. Select the Debug Flags tab.
3. Select Debug Flags.
4. To generate a log for a specific problem, specify the debug logging level on this tab page. Otherwise, you can leave the check boxes on this tab page deselected.

Setting Debug Logging Levels by Using the OID Control Utility

To set debug logging levels by using the OID Control Utility, restart the Oracle directory server using the -debug flag for an LDAP server, and the -d flag for the replication server. Use the debug level number based on Table 14–3 on page 14-6.

Because debug levels are additive, you need to add the numbers representing the functions that you want to activate, and use the sum of those in the command-line option.

By default, debug logging is turned off. To turn it on, modify the directory-specific entry (DSE) attribute orcldebugflag to the level you want. You can configure debug levels to one of the following levels.

To see debug log files generated by the OID Control Utility, navigate to $ORACLE_HOME/ldap/log.

Table 14–3 provides the complete list of debug logging levels.
Using Debug Logging

For example, to trace search filter processing (512) and active connection management (256), enter 768 as the debug level (512 + 256 = 768) as follows:

oidctl server=oidldapd instance=1 flags='-debug 768' restart
oidctl server=oidrepld instance=1 flags='-h my_host -p 389 -d 768' restart

This example restarts both the Oracle directory server as well as the Oracle directory replication server with the debugging flags.

Setting the Operation Debug Dimension

To make logging more focused, use the debug dimensions in conjunction with the debug levels. For example, to limit logging to particular directory server operations, specify the debug dimension to those operations.

Table 14–4 shows these dimensions.

Table 14–4 Debug Dimension Values for LDAP Operations

<table>
<thead>
<tr>
<th>Operation Debug Dimension Value</th>
<th>Provides Information Regarding</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ldapbind</td>
</tr>
<tr>
<td>2</td>
<td>ldapunbind</td>
</tr>
<tr>
<td>4</td>
<td>ldapadd</td>
</tr>
<tr>
<td>8</td>
<td>ldapdelete</td>
</tr>
</tbody>
</table>

For example, to trace search filter processing (512) and active connection management (256), enter 768 as the debug level (512 + 256 = 768) as follows:

oidctl server=oidldapd instance=1 flags='-debug 768' restart
oidctl server=oidrepld instance=1 flags='-h my_host -p 389 -d 768' restart

This example restarts both the Oracle directory server as well as the Oracle directory replication server with the debugging flags.

Table 14–3 Debug Logging Levels

<table>
<thead>
<tr>
<th>Logging Level Value</th>
<th>Provides Information Regarding</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Heavy trace debugging</td>
</tr>
<tr>
<td>128</td>
<td>Debug packet handling</td>
</tr>
<tr>
<td>256</td>
<td>Connection management, related to network activities</td>
</tr>
<tr>
<td>512</td>
<td>Search filter processing</td>
</tr>
<tr>
<td>1024</td>
<td>Entry parsing</td>
</tr>
<tr>
<td>2048</td>
<td>Configuration file processing</td>
</tr>
<tr>
<td>8192</td>
<td>Access control list processing</td>
</tr>
<tr>
<td>491520</td>
<td>Log of communication with the back end - that is with the database</td>
</tr>
<tr>
<td>524288</td>
<td>Schema related operations</td>
</tr>
<tr>
<td>4194304</td>
<td>Replication specific operations</td>
</tr>
<tr>
<td>8388608</td>
<td>Log of entries, operations and results for each connection</td>
</tr>
<tr>
<td>16777216</td>
<td>Trace function call arguments</td>
</tr>
<tr>
<td>67108864</td>
<td>Number and identity of clients connected to this server</td>
</tr>
<tr>
<td>117440511</td>
<td>All possible operations/data</td>
</tr>
<tr>
<td>134217728</td>
<td>All Java plug-in debug messages and internal server messages related to the Java plug-in framework</td>
</tr>
<tr>
<td>268435456</td>
<td>All messages passed by a Java plug-in using the ServerLog object.</td>
</tr>
<tr>
<td>402653184</td>
<td>Both of the above</td>
</tr>
</tbody>
</table>

For example, to trace search filter processing (512) and active connection management (256), enter 768 as the debug level (512 + 256 = 768) as follows:

oidctl server=oidldapd instance=1 flags='-debug 768' restart
oidctl server=oidrepld instance=1 flags='-h my_host -p 389 -d 768' restart

This example restarts both the Oracle directory server as well as the Oracle directory replication server with the debugging flags.

Table 14–4 Debug Dimension Values for LDAP Operations

<table>
<thead>
<tr>
<th>Operation Debug Dimension Value</th>
<th>Provides Information Regarding</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>ldapbind</td>
</tr>
<tr>
<td>2</td>
<td>ldapunbind</td>
</tr>
<tr>
<td>4</td>
<td>ldapadd</td>
</tr>
<tr>
<td>8</td>
<td>ldapdelete</td>
</tr>
</tbody>
</table>
You can set the debug operation dimension by using either Oracle Directory Manager or ldapmodify.

**Setting the Operation Debug Dimension by Using Oracle Directory Manager**

To set the operation debug dimension:

1. In the navigator pane, expand Oracle Internet Directory Servers and select a server instance. The group of tab pages for that server appear in the right pane.
2. Select the Debug Flags tab.
3. Select Debug Operation Flag.

By default, all operations are selected. To generate a log for a specific operation, select the corresponding operation. You can select more than one operation.

**Setting the Operation Debug Dimension by Using ldapmodify**

To log more than one operation, add the values of their dimensions. For example, if you want to trace ldapbind (1), ldapadd (4) and ldapmodify (16) operations, then create an LDIF file setting the orcldebugop attribute to 21 (1 + 4 + 16 = 21). The LDIF file is as follows:

```
dn:
changetype:modify
replace:orcldebugop
orcldebugop:21
```

To load this file, enter:

```
ldapmodify -h host_name -p port_number -f file_name
```

**Force Flushing the Trace Information to a Log File**

To minimize the performance overhead in I/O operations, debug messages are flushed to the log file periodically instead of every time a message is logged by the directory server. Writing to the log file is performed when one of the following occur:

- An LDAP operation completes
- A high priority message is logged
- The trace messages buffer is full

You can, however, view the trace messages in the log file as they are logged without having to wait for the periodic flush. To do this, set the DSA configuration attribute orcldebugforceflush to 1. Do this by using ldapmodify as shown in the following example.
Using the Audit Log

Example 14–1  Enabling Force Flushing

To enable force flushing by using ldapmodify:

1. Create an LDIF file as follows:

   ```
   dn: cn=dsaconfig,cn=configsets,cn=oracle internet directory
   changetype: modify
   replace: orcldebugforceflush
   orcldebugforceflush: 1
   ```

2. Load this file by entering the following:

   ```
   ldapmodify -h host_name -p port_number -f file_name
   ```

Note:

- When force flushing is enabled, the format of the trace message object for every operation becomes fragmented.
- By default, force flushing is inhibited. After you have flushed the necessary information to the log file, you should disable force flushing.

See Also:  "Oracle Identity Management LDAP Attribute Reference" in Oracle Identity Management User Reference for information about the orcldebugforceflush attribute

Using the Audit Log

The audit log records critical events on the Oracle directory server that are important from both a security and an operational point of view. Because the log generation depends on events on the directory server, you cannot create audit log entries. Only the directory server itself can create them.

The audit log is made up of regular directory entries, one entry for each event. You can query the audit log by using ldapsearch, and you can view the audit log entries by using Oracle Directory Manager.

By default, audit logging is disabled. To enable it, modify the directory-specific entry (DSE) attribute orclauditlevel to the level you want. You can configure audit levels to audit only selected events.

This section contains these topics:

- Structure of Audit Log Entries
- Position of Audit Log Entries in the DIT
- Auditable Events
- Setting the Audit Level
- Searching for Audit Log Entries
- Purging the Audit Log
Using the Audit Log

Structure of Audit Log Entries

Each audit log entry contains the orclAuditoc object class. Like all other structural object classes, orclAuditoc inherits from top. Table 14–5 lists and describes the attributes of the orclAuditoc object class.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>orclsequence</td>
<td>Used to create the name of the entry. The name is generated using a database sequence.</td>
</tr>
<tr>
<td>orcleventtype</td>
<td>Specifies the type of event that occurred. This is a cataloged attribute.</td>
</tr>
<tr>
<td>orcleventtime</td>
<td>Specifies the time at which the event occurred. This is formatted in UTC (Coordinated Universal Time). UTC is indicated by a z at the end of the value. For example, orcleventtime: 199811281010z</td>
</tr>
<tr>
<td>orcluserdn</td>
<td>Specifies the identity of the user who logged into the Oracle directory server to perform the operation. This attribute is cataloged.</td>
</tr>
<tr>
<td>orclopresult</td>
<td>Specifies the outcome of the operation. It states either SUCCESS if the operation succeeds, or the reason why the operation failed.</td>
</tr>
<tr>
<td>orclauditmessage</td>
<td>Specifies the textual message. This attribute is not cataloged.</td>
</tr>
<tr>
<td>objectclass</td>
<td>Contains the preset values top and orclAuditoc.</td>
</tr>
</tbody>
</table>

Note that the audit log entries do not become part of a regular search result set even though the search filter can satisfy the query criteria. For example, a search with the condition objectclass=top does not yield results from the auditlog entries. Only a search with cn=auditlog as the base of the search can find audit log entries.

See Also:
- "Auditable Events" on page 14-10 for a listing of audit levels
- "Setting the Audit Level" on page 14-11 for instructions on specifying the audit level
- "Searching for Audit Log Entries by Using Oracle Directory Manager" on page 14-12
- "Searching for Audit Log Entries by Using ldapsearch" on page 14-13
- The ldapdelete command-line tool reference in Oracle Identity Management User Reference

Note: By default, the attributes orcleventtype and orcluserdn are indexed at installation of Oracle Internet Directory. If you drop the indexes from these attributes, you cannot search for them. To re-create the index for these attributes, use the Catalog Management tool. See "Indexing an Attribute by Using Oracle Directory Manager" on page 11-13.
See Also:

- The catalog command-line tool reference in Oracle Identity Management User Reference for information about cataloged attributes
- "Object Class Types" on page 3-12 for a description of top

Position of Audit Log Entries in the DIT

The audit log container is part of the DSE. As shown in Figure 14–1, it holds its entries as children organized according to the orclsequence attribute.

Figure 14–1 Sample Audit Log in DSE

Auditable Events

Table 14–6 shows the auditable events and their audit levels. The third column, Audit Levels, contains hexadecimal values. You can audit more than one event by adding their corresponding values found in this column.

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
<th>Audit Levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Super user login</td>
<td>Super user bind to the server (successes or failures)</td>
<td>0x0001</td>
</tr>
<tr>
<td>Schema element add/replace</td>
<td>Addition of a new schema element (successes or failures)</td>
<td>0x0002</td>
</tr>
<tr>
<td>Schema element delete</td>
<td>Deletion of a schema (successes or failures)</td>
<td>0x0004</td>
</tr>
<tr>
<td>Bind</td>
<td>Unsuccessful bind cases</td>
<td>0x0008</td>
</tr>
<tr>
<td>Access violation</td>
<td>Access denied by access control policy point (ACP)</td>
<td>0x0010</td>
</tr>
<tr>
<td>directory-specific entry (DSE) modification</td>
<td>Changes to a DSE (successes or failures)</td>
<td>0x0020</td>
</tr>
<tr>
<td>Replication login</td>
<td>Replication server authentication (successes or failures)</td>
<td>0x0040</td>
</tr>
<tr>
<td>ACI modification</td>
<td>Changes to an access control list (ACL)</td>
<td>0x0080</td>
</tr>
<tr>
<td>User password modification</td>
<td>Modification of user password attribute</td>
<td>0x0100</td>
</tr>
<tr>
<td>Add</td>
<td>Idapadd operation (successes or failures)</td>
<td>0x0200</td>
</tr>
</tbody>
</table>
Setting the Audit Level

The setting for the DSE attribute orclauditlevel indicates the current audit level. You can enable or disable the events described in the previous section. A value of 0 for this attribute, which is the default, disables auditing.

You can set the audit level by using either Oracle Directory Manager or ldapmodify. This section describes both methods.

Setting the Audit Level by Using Oracle Directory Manager

To set the audit level by using Oracle Directory Manager:

1. In the navigator pane, expand Oracle Internet Directory Servers and select the directory server instance.
2. In the right pane, select the Audit Mask Levels tab page. This tab page lists the auditable events described in Table 14–7.

### Table 14–7   Audit Mask Levels

<table>
<thead>
<tr>
<th>Audit Level</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Super user login</td>
<td>Super user bind to the server (successes or failures)</td>
</tr>
<tr>
<td>Schema element add/replace</td>
<td>Addition of a new schema element (successes or failures)</td>
</tr>
<tr>
<td>Schema element delete</td>
<td>Deletion of a schema (successes or failures)</td>
</tr>
<tr>
<td>Bind</td>
<td>Unsuccessful bind cases</td>
</tr>
<tr>
<td>Access violation</td>
<td>Access denied by ACP</td>
</tr>
<tr>
<td>DSE modification</td>
<td>Changes to DSE entry (successes or failures)</td>
</tr>
<tr>
<td>Replication login</td>
<td>Replication server authentication (successes or failures)</td>
</tr>
<tr>
<td>ACL modification</td>
<td>Changes to ACPs</td>
</tr>
<tr>
<td>User password modification</td>
<td>Modification of user password attribute</td>
</tr>
<tr>
<td>Add</td>
<td>ldapadd operation (successes or failures)</td>
</tr>
<tr>
<td>Delete</td>
<td>ldapdelete operation (successes or failures)</td>
</tr>
<tr>
<td>Modify</td>
<td>ldapmodify operation (successes or failures)</td>
</tr>
<tr>
<td>ModifyDN</td>
<td>ldapModifyDN operation (successes or failures)</td>
</tr>
</tbody>
</table>

3. Select the audit level you want to use.

Both successful and unsuccessful events are entered into the audit log if they are selected, except:

### Table 14–6   (Cont.) Auditable Events

<table>
<thead>
<tr>
<th>Event</th>
<th>Description</th>
<th>Audit Levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete</td>
<td>Idapdelete operation (successes or failures)</td>
<td>0x0400</td>
</tr>
<tr>
<td>Modify</td>
<td>Idapmodify operation (successes or failures)</td>
<td>0x0800</td>
</tr>
<tr>
<td>ModifyDN</td>
<td>ldapModifyDN operation (successes or failures)</td>
<td>0x1000</td>
</tr>
<tr>
<td>bind</td>
<td>Successful user bind cases</td>
<td>0x2000</td>
</tr>
</tbody>
</table>
Using the Audit Log

- Bind, which logs only unsuccessful bind attempts
- Access Violation, which logs only events in which access is denied by an ACP

4. Choose **Apply**.
5. Restart the directory server instance for the changes to take effect.

**See Also:** The `oidctl` command-line tool reference in *Oracle Identity Management User Reference* for instructions on how to restart the directory server

### Setting the Audit Level by Using `ldapmodify`

To audit more than one event, add the values of their audit masks. For example, suppose you want to audit the events in Table 14–8.

<table>
<thead>
<tr>
<th>Event</th>
<th>Audit Level</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Schema element delete</td>
<td>0x0004</td>
<td>4</td>
</tr>
<tr>
<td>DSE modification</td>
<td>0x0020</td>
<td>32</td>
</tr>
<tr>
<td>Add</td>
<td>0x0200</td>
<td>512</td>
</tr>
</tbody>
</table>

The total value of the audit levels is 548. The `ldapmodify` command would therefore look something like this:

```
ldapmodify -p port -h host << EOF
dn:
  changetype:modify
  replace: orclauditlevel
  orclauditlevel: 548
EOF
```

Restart the directory server instance after any changes are made to `orclauditlevel` for the changes to take effect.

**See Also:** The `oidctl` command-line tool reference in *Oracle Identity Management User Reference* for instructions on how to restart the directory server

### Searching for Audit Log Entries

You can search for audit log entries by using either Oracle Directory Manager or `ldapsearch`.

### Searching for Audit Log Entries by Using Oracle Directory Manager

To use Oracle Directory Manager to view audit log entries:

1. In the navigator pane, expand **Oracle Internet Directory Servers** and **directory server instance**.
2. Select **Audit Log Management**. The corresponding right pane appears.
3. In the **Max Results (entries)** field, type the maximum number of entries you want your search to retrieve. The default is 200. The directory server retrieves the number you specify, up to 1000.
4. In the **Max Search Time (seconds)** box, type the maximum number of seconds for the duration of your search. The value you enter here must be at least that of the default, namely, 25. The directory server searches for the amount of time you specify, up to one hour.

5. In the **Search Criteria** box, use the lists and text fields on the search criteria bar to focus your search.
   
a. From the list at the left end of the search criteria bar, select an attribute of the entry you want to search for. Because not all attributes are used in every entry, be sure that the attribute you specify actually corresponds to one in the entry that you are searching for. Otherwise, the search fails.
   
b. From the list in the middle of the search criteria bar, select a filter. These are described in Table A–45 on page A-31.
   
c. In the text box at the right end of the search criteria bar, type the value for the attribute you just selected. For example, if the attribute you selected was `cn`, you could type the particular common name you want to find.

6. To further refine your search, use the buttons in the **Search Criteria** box to enhance the search criteria bar. These are described in Table A–46 on page A-31.

7. Choose **Search**. The results of your search appear in the Distinguished Name box.

8. To view the properties of a particular audit log entry, select it in the **Distinguished Name** box, then choose to exploit the features of Oracle Internet Directory Server Manageability. The Audit Log Entry dialog box displays the properties for the audit log entry you selected.

   **See Also:** "Configuring the Display and Duration of Searches in **Oracle Directory Manager**" on page 5-6 for instructions on setting the number of entries to display in searches, and to set the time limit for searches

**Searching for Audit Log Entries by Using ldapsearch**

The DN for the audit log container is `cn=auditlog`. To search for audit log entries, perform a subtree or one-level search, with the container object `cn=auditlog` as the base of the search.

   **See Also:** The `ldapsearch` command-line tool reference in **Oracle Identity Management User Reference**

**Purging the Audit Log**

You can use bulkdelete to purge audit log objects under the container `cn=auditlog`. Run the following command:

```
bulkdelete connect="connect_string" basedn="cn=auditlog"
```

**Monitoring Oracle Internet Directory Servers**

Oracle Internet Directory Server Manageability enables you to monitor various types of information about Oracle Internet Directory servers. This section contains these topics:

- **Capabilities of Oracle Internet Directory Server Manageability**
- **Oracle Internet Directory Server Manageability Architecture and Components**
Capabilities of Oracle Internet Directory Server Manageability

The Oracle Internet Directory Server Manageability framework enables you to monitor the following directory server statistics:

- Server health statistics about LDAP request queues, memory, LDAP sessions, and database sessions. For example, you can view the number of active database sessions over a period of time. You can also view the total number of connections opened to Oracle Internet Directory server instances over a period of time.

- Performance statistics. Average latency in millisecond is provided for bind, compare, messaging search, and all search operations over a period of time.

- General statistics about specific server operations, such as add, modify, or delete. For example, you can view the number of directory server operations over a period of time.

- User statistics comprising successful and failed operations to the directory and the user performing each one. All LDAP operations are tracked for configured users. Also, the number of connections held by users at the ends of the statistics collection period are tracked.

- Critical events related to system resources and security—for example, occasions when a user provided the wrong password or had inadequate access rights to perform an operation. Other critical events include ORA errors other than expected errors including 1, 100 or 1403 and abnormal termination of the LDAP server.

- Security events tracking of users’ successful bind and userpassword compare operations, and userpassword compare failure.

- Status information of the directory server and the directory replication server—for example, the date and time at which the directory replication server was invoked.

- Status information of Oracle directory integration and provisioning server and the integration profiles—for example, the number of times that the directory integration server failed, or whether an integration profile is enabled.

See Also: The chapter on Oracle Directory Integration Platform concepts and components in Oracle Identity Management Integration Guide

Oracle Internet Directory Server Manageability Architecture and Components

The relationship between the various components of directory server manageability is explained in Figure 14–2 and the accompanying text in Table 14–9.
Monitoring Oracle Internet Directory Servers

Logging, Auditing, and Monitoring the Directory

Figure 14–2 Architecture of Oracle Internet Directory Server Manageability

Table 14–9 Components of Oracle Internet Directory Server Manageability

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Internet Directory</td>
<td>A directory server responds to directory requests from clients. It has four kinds of functional threads: controller, worker, dispatcher, and listener. It accepts LDAP requests from clients, processes them, and sends the LDAP response back to the clients. When you use the Oracle Internet Directory Server Manageability framework to set runtime monitoring, the four functional threads of the server record the specified information and store it in local memory. <strong>See Also:</strong> &quot;An Oracle Directory Server Instance&quot; on page 3-3 for a description of the directory server.</td>
</tr>
<tr>
<td>Memory Resident Storage</td>
<td>This is a local process memory. The Oracle Internet Directory Server Manageability framework assigns one each for statistics, tracing, and auditing. Each has its own separate data structure maintained in the local memory storage.</td>
</tr>
<tr>
<td>Low Priority Write Threads</td>
<td>These dedicated write threads differ from server functional threads in that they write server statistics, audit logging, and tracing information to the repository. To maintain reduced system overhead, their priorities are kept low.</td>
</tr>
<tr>
<td>External Monitoring Application</td>
<td>This module, which is proprietary and external to the server manageability framework, collects the gathered statistics through a standard LDAP interface with the directory server and stores it in its own repository.</td>
</tr>
</tbody>
</table>
Monitor the Oracle Internet Directory Server through the Manageability framework. This framework stores configuration parameters for server statistics, server tracing, and server auditing in the DSE root of the directory. To set periodicity, amount, and level of information to be gathered, specify the parameters.

**Location of Configuration Information for Oracle Internet Directory Server Manageability**

The Oracle Internet Directory Server Manageability framework stores configuration parameters for all three modules—server statistics, server tracing, and server auditing—in the DSE root of the directory. To specify periodicity, amount, and level of information to be gathered, set appropriate values for parameters.

**Account Used for Accessing Server Manageability Information**

The Oracle Internet Directory database account ODSSM is used to access server manageability information from the database. During installation, this account is given a randomly-generated password.

The credentials for this account, including the randomized password, are stored in the Enterprise Manager file targets.xml. The only way to change this account’s password is to use SQLPLUS. There is no support for changing this password with the oidpasswd tool or stored in a wallet. After altering this password in the database, you must also change it in the file targets.xml.

**Configuring Oracle Internet Directory Server Manageability**

To configure the Oracle Internet Directory Server Manageability framework, use ldapmodify to set positive integer values for various attributes in the root DSE:

- To enable health, general, and performance statistics, set the orclStatsFlag and orclStatsPeriodicity attributes.
To configure security events tracking, see “Configuring Security Events Tracking” on page 14-17.

To enable user statistics:
- Set the orclStatsLevel attribute to 1
- Set the orclStatsPeriodicity attribute

---

**Note:** When you are collecting statistics for Grid Control, set orclStatsPeriodicity to be the same as the collection periodicity of the Enterprise Manager agent, which is 10 minutes by default.

---

To configure users for statistics collection, see "Configuring a User for Connection and Operation Statistics Collection" on page 14-18.

To enable critical events, set the OrclEventLevel attribute. To configure critical events, see “Configuring Critical Events” on page 14-19.

To enable events other than super user, proxy user, and replication administrator login:
- Set the OrclEventLevel attribute to the appropriate value
- Set the orclStatsFlag to 1

**See Also:** "Oracle Identity Management LDAP Attribute Reference," in Oracle Identity Management User Reference for information about each of the attributes you set when using Oracle Internet Directory Server Manageability

For example, to enable the Oracle Internet Directory Server Manageability framework, you create an LDIF file that looks like this:

```plaintext
dn: changetype: modify
replace: orclstatsflag
orclstatsflag:1
```

To upload this file, enter the following command:

```
ldapmodify -h host -p port_number -D bind_DN -w bind_DN_password -f file_name
```

where the bind DN authorized to perform server manageability configuration is cn=emd admin,cn=oracle internet directory.

**See Also:** Online help for Oracle Enterprise Manager 10g Application Server Control Console for more information about monitoring and managing Oracle Internet Directory servers by using Oracle Internet Directory Server Manageability

### Configuring Security Events Tracking

To configure security events tracking, set the DSA Config Attribute orcloptracklevel by using either the command line or Oracle Directory Manager. This attribute is located in cn=dsaconfig,cn=configsets,cn=oracle internet directory. Table 14–10 lists the values of the DSA configuration attribute orcloptracklevel to configure different levels of bind and compare information collection.
The metrics recorded by each orcloptracklevel value are listed in the following table:

### Table 14–10  Values of the DSA Configuration Attribute orcloptracklevel

<table>
<thead>
<tr>
<th>orcloptracklevel</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bind DN only</td>
</tr>
<tr>
<td>2</td>
<td>Bind DN and IP address</td>
</tr>
<tr>
<td>4</td>
<td>Compare DN only</td>
</tr>
<tr>
<td>8</td>
<td>Compare DN and IP address</td>
</tr>
<tr>
<td>16</td>
<td>Compare DN, IP address and failure details</td>
</tr>
</tbody>
</table>

### Table 14–11  Metrics Recorded by Each orcloptracklevel Value

<table>
<thead>
<tr>
<th>Configuration</th>
<th>Metrics Recorded</th>
</tr>
</thead>
<tbody>
<tr>
<td>DN only</td>
<td>Date and time stamp</td>
</tr>
<tr>
<td></td>
<td>EID of DN performing the operation</td>
</tr>
<tr>
<td></td>
<td>Success counts</td>
</tr>
<tr>
<td></td>
<td>Failure counts</td>
</tr>
<tr>
<td>DN and IP address</td>
<td>All metrics listed under DN only</td>
</tr>
<tr>
<td></td>
<td>Source IP Address</td>
</tr>
<tr>
<td>DN, IP address and failure details</td>
<td>All metrics listed under DN and IP address</td>
</tr>
<tr>
<td></td>
<td>Distinct success counts</td>
</tr>
<tr>
<td></td>
<td>Distinct failure counts</td>
</tr>
<tr>
<td></td>
<td>Failure details for each DN performing password compare from an IP Address:</td>
</tr>
<tr>
<td></td>
<td>Date and time stamp</td>
</tr>
<tr>
<td></td>
<td>Source IP Address</td>
</tr>
<tr>
<td></td>
<td>EID of DN whose password is compared</td>
</tr>
<tr>
<td></td>
<td>Failure counts</td>
</tr>
</tbody>
</table>

Two attributes, orcloptracknumelemcontainers and orcloptrackmaxtotalsize, allow you to tune memory used for tracking security events. See “Tuning Security Event Tracking” on page 25-6.

### Configuring a User for Connection and Operation Statistics Collection

To configure a user for connection statistics collection, add the user’s DN to the DSA configset entry’s multivalued attribute orclstatsdn (DN: cn=dsaconfig, cn=configsets, cn=oracle internet directory) by using the ldapmodify command line tool. You can do this by using Oracle Directory Manager or by using the command line. To configure a user by using Oracle Directory Manager, do the following:

1. In the navigator pane, expand Oracle Internet Directory Servers and select the directory server instance.
2. In the right pane, select the Query Optimization tab.
3. Add the user’s DN to the Monitored Users’ DNs field on the Query Optimization tab page.
To configure a user by using the command line, add the user’s DN to the DSA Configset entry’s multivalued attribute orclstatsdn (DN: cn=dsaconfig,cn=configsets,cn=oracle internet directory) by using the ldapmodify command line tool. For example:

```
ldapmodify -h host -p port <<EOF
dn: cn=dsaconfig,cn=configsets,cn=oracle internet directory
changetype:modify
add: orclstatsdn
orclstatsdn: userDN
EOF
```

**Configuring Critical Events**

To configure critical events, use ldapmodify to set the OrclEventLevel attribute to one or more of the event levels listed in Table 14–12.

<table>
<thead>
<tr>
<th>Level Value</th>
<th>Critical Event</th>
<th>Information It Provides</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Super user login</td>
<td>Super uses bind (successes or failures)</td>
</tr>
<tr>
<td>2</td>
<td>Proxy user login</td>
<td>Proxy user bind (failures)</td>
</tr>
<tr>
<td>4</td>
<td>Replication login</td>
<td>Replication bind (failures)</td>
</tr>
<tr>
<td>8</td>
<td>Add access</td>
<td>Add access violation</td>
</tr>
<tr>
<td>16</td>
<td>Delete access</td>
<td>Delete access violation</td>
</tr>
<tr>
<td>32</td>
<td>Write access</td>
<td>Write access violation</td>
</tr>
<tr>
<td>64</td>
<td>ORA 3113 error</td>
<td>ORA-3113 Error</td>
</tr>
<tr>
<td>128</td>
<td>ORA 3114 error</td>
<td>ORA-3114 Error</td>
</tr>
<tr>
<td>256</td>
<td>ORA 28 error</td>
<td>ORA-28 Error</td>
</tr>
<tr>
<td>512</td>
<td>ORA error</td>
<td>ORA errors other an expected 1, 100, or 1403</td>
</tr>
<tr>
<td>1024</td>
<td>Oracle Internet server</td>
<td></td>
</tr>
<tr>
<td></td>
<td>termination count</td>
<td></td>
</tr>
<tr>
<td>2047</td>
<td>All critical events</td>
<td></td>
</tr>
</tbody>
</table>

**Purging of Audit and Statistics Entries**

Obsolete audit and statistics entries are removed from Oracle Internet Directory by the Oracle Internet Directory purge tool, described in Chapter 26, "Garbage Collection in Oracle Internet Directory".

**Viewing Oracle Internet Directory Server Manageability Information**

This section contains these topics:

- Viewing Information with the oiddiag Tool
- Viewing Information with the Oracle Identity Management Grid Control Plug-in
- Viewing Information with the Oracle Enterprise Manager 10g Application Server Control Console
Monitoring Oracle Internet Directory Servers

Viewing Information with the oiddiag Tool
Reports for all the statistics can be viewed using the oiddiag tool, as follows:

oiddiag audit_report=true outfile=file_name

See Also:
- The oiddiag command tool reference in Oracle Identity Management User Reference.
- The chapter about administration tools in the Oracle Application Server Administrator’s Guide

Viewing Information with the Oracle Identity Management Grid Control Plug-in
Several new 10g (10.1.4.0.1) features can be viewed by using the Oracle Identity Management Grid Control Plug-in. This new Oracle Enterprise Manager interface is described in Oracle Identity Management Infrastructure Administrator’s Guide.

See Also:
- The Oracle Identity Management Grid Control Plug-in chapter in Oracle Identity Management Infrastructure Administrator’s Guide.
- Online help for Oracle Enterprise Manager 10g Application Server Control Console
- Online help for the Oracle Identity Management Grid Control Plug-in

Viewing Information with the Oracle Enterprise Manager 10g Application Server Control Console
To view many of the features of Oracle Internet Directory Server Manageability, you use Oracle Enterprise Manager 10g Application Server Control Console as explained in this section.

Note:
The Application Server Control Console does not display the port status information for Oracle directory servers running only in SSL mode.

See Also: Oracle Application Server Administrator’s Guide for information about stopping and starting Oracle Enterprise Manager 10g Application Server Control Console

Enabling Information Collection by Using Oracle Enterprise Manager 10g Application Server Control Console
To enable information collection by using Oracle Enterprise Manager 10g Application Server Control Console:

1. In the Oracle Internet Directory main window, select LDAP Metrics. This displays the LDAP Diagnostic Collection Configuration page.
2. Check Collect Metrics.
3. Select Interval.
4. Enter the required password.
5. Choose **Apply**.

---

**Note:** To enable critical events, use ldapmodify to set the attribute `orclEventLevel` to the appropriate value.

---

**Starting a New Directory Server Instance by Using Oracle Enterprise Manager 10g Application Server Control Console**

To start a server:

1. In the Oracle Internet Directory main window, choose **Start New Instance**. The Start a New LDAP Server Instance Window displays the fields in **Table 14–13**.

**Table 14–13  Fields in the Start a New LDAP Server Instance Window of the Application Server Control Console**

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set Number</td>
<td>The configuration set number for the directory server instance</td>
</tr>
<tr>
<td>Default Port</td>
<td>The default port number for the directory server instance</td>
</tr>
<tr>
<td>Port Available</td>
<td>Indicator of whether the default port is available</td>
</tr>
<tr>
<td>Maximum Database Connections</td>
<td>The number of database connections this directory instance can accommodate</td>
</tr>
<tr>
<td>Server Processes</td>
<td>The number of server processes</td>
</tr>
<tr>
<td>Port Number</td>
<td>The port number you assign to the directory server instance if the default port number is not used</td>
</tr>
</tbody>
</table>

2. In the **Set Number** column, select the configuration set you want to use.

   If the default port is not available, then, in the **Port Number** column, specify a port number.

3. Choose **Start**.

**Stopping a Directory Server Instance by Using Oracle Enterprise Manager 10g Application Server Control Console**

To stop a directory server instance:

1. In the Oracle Internet Directory main window, in the **LDAP Instances** section, select the directory server instance you want to stop.

2. Choose **Stop**.

**Restarting a Directory Server Instance by Using Oracle Enterprise Manager 10g Application Server Control Console**

To restart a directory server instance:

1. In the Oracle Internet Directory main window, in the **LDAP Instances** section, select the server you want to restart.

2. Choose **Restart**. The Restart an LDAP Server Instance window displays the fields listed in **Table 14–14**.

**Table 14–14  Fields in the Restart an LDAP Server Instance Window of the Application Server Control Console**

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set Number</td>
<td>The configuration set number for the directory server instance</td>
</tr>
<tr>
<td>Default Port</td>
<td>The default port number for the directory server instance</td>
</tr>
<tr>
<td>Port Available</td>
<td>Indicator of whether the default port is available</td>
</tr>
</tbody>
</table>
3. Select a configuration. If the default port is not available, then, in the Port Number column, enter a port number.

4. Choose Start.

**Viewing Directory Server Activities by Using Oracle Enterprise Manager 10g Application Server Control Console**

To view directory server activities information:

1. In the Directory Server main window, select the directory server instance whose information you want to view.

2. Choose View Load. The LDAP Load window appears.

3. From the Select Load Characteristics list, select the information that you want to view about this instance. The options are:

   - **LDAP Repository Database Sessions**: Selecting this option displays two graphs, one for open database sessions, the other for active database sessions at the end of the specified time period of statistics collection.

   - **Response Time vs. LDAP Operations**: Selecting this option displays two graphs. The first shows the average LDAP operation response time over the course of the specified time period of statistics collection. The other shows the number of operations in progress at the end of that period.

   - **Active LDAP Sessions vs. New LDAP Sessions**: Selecting this option displays two graphs. The first shows the number of active LDAP sessions, that is, those that remain open at the end of the specified time period of statistics collection. The second shows new LDAP sessions, that is, those that are opened over the course of the specified time period of statistics collection.

4. When you have made your selection, choose Go.

**Viewing Directory Server Operations by Using Oracle Enterprise Manager 10g Application Server Control Console**

You can view directory server operations over the course of the specified time period of statistics collection by using Application Server Control Console. To do this:

1. In the Directory Server main window, select the directory server instance whose information you want to view.

2. Choose View Operations. This displays charts for all of the LDAP operations. Click any chart to see a larger image of it.
This chapter tells how to backup and restore both small and large directories. It contains these topics:

- Backing Up and Restoring a Small Directory or Specific Naming Context
- Backing Up and Restoring a Large Directory

**Backing Up and Restoring a Small Directory or Specific Naming Context**

To backup and restore a small directory or specific naming context in directory, do the following:

1. Backup the node by using the ldifwrite utility. Enter this command:
   
   ```
   ldifwrite connect="connect_string" basedn="naming_context" file="backup.ldif"
   ```

2. Start the directory server on the new node by entering this command:
   
   ```
   oidctl connect=connect_string server=oidldapd instance=1 flags=' -p port_number' start
   ```

3. Load data into the new node by using the ldapaddmt utility. Enter this command:
   
   ```
   bulkload connect="connect_string" check="TRUE" generate="TRUE" load="TRUE" restore="TRUE" append="TRUE" file="/complete_path/backup.ldif"
   ```

**Note:** If you back up data from an earlier version of Oracle Internet Directory, such as 10g Release 2 (10.1.2.0.2), then restore it on a node running 10g (10.1.4.0.1), you must update the password policy entries as described in "Password Policy and Fan-out Replication" on page 30-32.

**Backing Up and Restoring a Large Directory**

For instructions on backing up and restoring a large directory, see *Oracle Application Server Administrator’s Guide*. 

Backup and Restoration of a Directory
Part III
Directory Security

This part explains how to:

- Secure data within the directory
- Establish access controls for administering applications in enterprises and hosted environments
- Establish and manage policies governing passwords
- Manage password verifiers used to authenticate users to other Oracle components
- Store data for users, groups, and services in one repository, and delegate the administration of that data to various administrators

It contains these chapters:

- Chapter 16, "Directory Security Concepts"
- Chapter 17, "Secure Sockets Layer (SSL) and the Directory"
- Chapter 18, "Directory Access Control"
- Chapter 19, "Password Policies in Oracle Internet Directory"
- Chapter 20, "Directory Storage of Password Verifiers"
- Chapter 21, "Delegation of Privileges for an Oracle Technology Deployment"
Oracle Internet Directory is a key element of the Oracle Identity Management Infrastructure. This enables you to deploy multiple Oracle components to work against a shared instance of Oracle Internet Directory and associated infrastructure pieces. This sharing allows an enterprise to simplify security management across all applications.

In addition to the role it plays in the Oracle Identity Management infrastructure, Oracle Internet Directory provides many powerful features for protecting information.

This chapter gives a conceptual overview of Oracle Internet Directory security features. It contains these topics:

- Data Integrity and Oracle Internet Directory
- Data Privacy and Oracle Internet Directory
- Authorization in Oracle Internet Directory
- Authentication in Oracle Internet Directory
- Protection of User Passwords for Directory Authentication
- Password Policies in Oracle Internet Directory
- Authentication by Using Simple Authentication and Security Layer (SASL)

**Data Integrity and** Oracle Internet Directory

Oracle Internet Directory ensures that data has not been modified, deleted, or replayed during transmission by using Secure Sockets Layer (SSL). SSL 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.

*See Also:* Chapter 17, "Secure Sockets Layer (SSL) and the Directory" for more information about SSL

**Data Privacy and** Oracle Internet Directory

Data privacy is a concern both during transmission and after the data is received. This section explains how Oracle Internet Directory protects data in these two situations. This section contains the following topics:

- Privacy During Data Transmission
- Privacy of Retrieved Sensitive Attributes
Privacy During Data Transmission

Oracle Internet Directory 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. Specifically, Oracle Internet Directory supports two levels of encryption available through SSL:

- **DES40**
  
  The DES40 algorithm, available internationally, is a variant of DES in which the secret key is preprocessed to provide 40 effective key bits. It is designed for use by customers outside the USA and Canada who want to use a DES-based encryption algorithm. This feature gives commercial customers a choice in the algorithm they use, regardless of their geographic location.

- **RC4_40**
  
  Oracle has obtained license to export the RC4 data encryption algorithm with a 40-bit key size to virtually all destinations where other Oracle products are available. This makes it possible for international corporations to safeguard their entire operations with fast cryptography.

See Also:  Chapter 17, "Secure Sockets Layer (SSL) and the Directory" for more information about SSL

Privacy of Retrieved Sensitive Attributes

Oracle Internet Directory stores sensitive attributes in an encrypted format. Examples of sensitive attributes are: orclpasswordattribute, orclrevpwd, the plug-in attribute orclplugininsuredflexfield and the server chaining attribute orclIDSCExtPassword.

The orcl-dataprivacymode attribute controls whether these attributes are encrypted when the data is received. When orcl-dataprivacymode is enabled, the sensitive attributes are encrypted. When privacy mode is disabled, the sensitive data is returned in the clear.

Privacy mode is disabled by default. That is, the value of orcl-dataprivacymode is 0. To provide security protection, you must enable privacy mode by changing the value of orcl-dataprivacymode from 0 to 1.

To determine the value of orcl-dataprivacymode, perform the following search:

```bash
$ORACLE_HOME/bin/ldapsearch -h host -p port -D cn=orcladmin -w password -b "cn=dsaconfig,cn=configsets,cn=oracle internet directory" -s base "objectclass=*" orcl-dataprivacymode
```

To enable privacy mode, use an LDIF file containing the following entries:

```bash
dn: cn=dsaconfig,cn=configsets,cn=oracle internet directory
changetype: modify
replace: orcl-dataprivacymode
orcl-dataprivacymode: 1
```

Load the LDIF file with a command line similar to this:

```bash
$ORACLE_HOME/bin/ldapmodify -h host -p port -D cn=orcladmin -w password -v -f LDIF_file_name
```
Authorization in Oracle Internet Directory

Authorization is the permission given to a user, program, or process to access an object or set of objects. When directory operations are attempted within a directory session, the directory server ensures that the user has the permissions to perform those operations. If the user does not have the permissions, then the directory server disallows the operation. The directory server protects directory data from unauthorized operations by directory users by using access control information.

Access control information is the directory metadata that captures the administrative policies relating to access control. This information is stored in Oracle Internet Directory as user-modifiable operational attributes, each of which is called an access control item (ACI).

Typically, a list of these ACI attribute values, called an access control list (ACL), is associated with directory objects. The attribute values on that list represent the permissions that various directory user entities (or subjects) have on a given object.

An ACI consists of:

- The object to which you are granting access
- The entities or subjects to whom you are granting access
- The kind of access you are granting

Access control policies can be prescriptive, that is, their security directives can be set to apply downward to all entries at lower positions in the directory information tree (DIT). The point from which such an access control policy applies is called an access control policy point (ACP).

ACIs are represented and stored as text strings in the directory. These strings must conform to a well-defined format, called the ACI directive format. Each valid value of an ACI attribute represents a distinct access control policy.

The following features of directory access control can be used by applications running in a hosted environment.

- Prescriptive access control
  Enables the service provider to specify access control lists (ACLs) for a collection of directory objects, instead of having to state the policies for each individual object. This feature simplifies the administration of access control, especially in large directories where many objects are governed by identical or similar policies.

- Hierarchical access control administration model
  Enables the service provider to delegate directory administration to hosted companies. The realm could in turn delegate further if necessary.

- Administrative override control for delegated domains
  Enables the service provider to perform diagnosis and recovery from unintentional account lockout or accidental security exposure.

- Dynamic evaluation of access control entities
  Enables subtree administrators to identify both subjects and objects in terms of their namespace and their association with other objects in the directory. For example, the administrator of one realm can allow only a user’s manager to update that user’s salary attribute. The administrator of another realm can establish and enforce a different policy regarding salary attributes.
Authentication in Oracle Internet Directory

Authentication in Oracle Internet Directory

Authentication is the process by which the 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. Thus every session has an associated user identity.

To verify the identities of users, hosts, and clients, Oracle Internet Directory enables three general kinds of authentication, and these are described in these topics:

- Direct Authentication
- Indirect Authentication
- External Authentication

Direct Authentication

This section describes the three kinds of direct authentication available within Oracle Internet Directory, and about how SASL-enabled clients authenticate to a directory server. The three kinds of direct authentication options are:

- Anonymous authentication

  When users authenticate anonymously, they simply leave the user name and password fields blank when they log in. Each anonymous user then exercises whatever privileges are specified for anonymous users.

- Simple authentication

  When using simple authentication, the client identifies itself to the server by means of a DN and a password that are not encrypted when sent over the network.

- Authentication by using Simple Authentication and Security Layer (SASL)

  This is a method for adding authentication support to connection-based protocols. To use SASL, a protocol includes a command for identifying and authenticating a user to a server and for optionally negotiating protection of subsequent protocol interactions. If the use of SASL is successfully negotiated, then a security layer is inserted between the protocol and the connection.

Oracle Internet Directory supports two authentication mechanisms with SASL:

- Digest MD5: The required authentication mechanism within LDAP Version 3 (RFC 2829). It uses the MD5 hash function to convert a message of any length to a 128 bit message digest that can be used as a verifier for client/server authentication.

- External authentication: Mechanism using SSL mutual authentication. In this case, the client, in lieu of a user name and password, authenticates to the server by means of a certificate, token, or some other device. Certificate authentication can take the following forms:

  * Exact match: the subject DN in the client certificate is compared with the user DN in the directory. If the two values match, a bind occurs.

  * Certificate hash: The client certificate is hashed and is then compared with the hashed value of the certificate stored in the directory. If the two values match and only one DN is associated with the pair, a bind occurs. If two or more DNs are associated, an error will be returned because certificate hash and user DN is an n-to-1 mapping and not a 1-to-n mapping. That is, you
can have many certificates associated with one DN, but only one DN associated with a certificate.

* Exact match/certificate hash: An exact-match search is performed first. If this search yields nothing, a certificate hash is performed.

To choose one of these methods, edit the DSA configuration parameter orclpkimatchingrule as prescribed in “Oracle Identity Management LDAP Attribute Reference” in Oracle Identity Management User Reference. (For authentication, an orclpkimatchingrule value of 3 or 4 is equivalent to a value of 2.)

Notes:

- The introduction in 10g (10.1.4.0.1) of a certificate hash value requires that user certificates be upgraded from earlier releases. To learn how to upgrade certificates, see the upgradecert.pl command-line tool reference in Oracle Identity Management User Reference.

- You can search for the binary attribute usercertificate. To learn how to conduct a search, see Appendix G, "Searching the Directory for User Certificates".

See Also:

- Authentication by Using Simple Authentication and Security Layer (SASL) on page 16-8

- The Web site of the Internet Engineering Task Force (IETF) at http://www.ietf.org for the following RFCs: RFC 2829, which specifies SASL Digest-MD5 as the required authentication mechanism for LDAP Version 3 servers; RFC 2831, which describes the Digest-MD5 mechanism; RFC 2617, which describes the HTTP Digest authentication mechanism on which SASL Digest-MD5 is based

Indirect Authentication

Indirect authentication occurs through any entity that has credentials in the directory—for example, an application such as the Oracle Internet Directory Self-Service Console, or a middle tier such as a firewall or a RADIUS server. The application or middle tier becomes a proxy user. A proxy user has the privilege to impersonate an end user, performing on that user's behalf those operations for which that user has privileges.

Figure 16–1 and the accompanying text explain how indirect authentication takes place.
Indirect authentication takes place as follows:

1. The end user sends to the application or middle tier a request containing a query to Oracle Internet Directory. The application or middle tier authenticates the end user.

2. The application or middle tier binds to the directory.

3. The application or middle tier performs a second bind, this time using the DN of the end user. It does not enter the end user's password.

4. The directory server recognizes this second bind as an attempt by the application or middle tier to switch to the end user's identity. It trusts the authentication granted to the end user by the application or middle tier, but must verify that the application or middle tier has the right to be the proxy for this user. It checks to see whether the ACP governing the end user entry gives this application or middle tier the proxy right for this end user.
   - If the end user entry does give the application or middle tier the necessary proxy right, then the directory server changes the authorization identity to that of the end user. All subsequent operations occur as if that end user had connected directly to the server and had been directly authenticated.
   - If the end user entry does not give the application or middle tier the necessary proxy right, then the directory server returns an "Insufficient Access" error message.

See Also: Operations: What Access Are You Granting? on page 18-8

The directory server can, in the same session, authenticate and authorize other end users. It can also switch the session from the end user to the application or middle tier that opened the session.

To close the session, the application or middle tier sends an unbind request to the directory server.

For example, suppose you have:

- A middle tier that binds to the directory as cn=User1, which has proxy access on the entire directory
- An end user that can bind to the directory as cn=User2
When this end user sends to the application or middle tier a request containing a query to the directory, the application or middle tier authenticates the end user. The middle tier service then binds to the directory by using its own identity, \texttt{cn=User1}, then performs a second bind, this time by using only the DN of the end user, \texttt{cn=User2}. The Oracle directory server recognizes this second bind as an attempt by the proxy user to impersonate the end user. After the directory server verifies that \texttt{cn=User1} has proxy access, it allows this second bind to succeed. It does not require any further validation of the end-user DN, such as a password. For the rest of the session, all LDAP operations are access-controlled as if \texttt{cn=User2} were performing them.

If one user is being serviced by an application, and another user subsequently requests a service of that same application, then the application can establish a new connection and proceed as previously described without disrupting that prior session. If, however, no prior user is still being serviced, then the existing established connection can be re-used again and again without the need for a new connection.

**External Authentication**

Perhaps your enterprise stores user security credentials in a repository other than Oracle Internet Directory—for example, a database or another LDAP directory. With Oracle Internet Directory external authentication and password modification plug-ins, you can use these credentials for user authentication to Oracle components. You do not need to store the credentials in Oracle Internet Directory and then worry about keeping them synchronized.

*See Also:* Chapter 34, "Setting Up the Customized External Authentication Plug-in"

**Protection of User Passwords for Directory Authentication**

Oracle Internet Directory can protect a user's directory password by storing it in the `userPassword` attribute as a one-way hashed value. You select the hashing algorithm you want to use. Storing passwords as one-way hashed values—rather than as encrypted values—more fully secures them because a malicious user can neither read nor decrypt them.

*See Also:* Chapter 20, "Directory Storage of Password Verifiers"

**Password Policies in Oracle Internet Directory**

A password policy is a set of rules governing how passwords are used. When a user attempts to bind to the directory, the directory server ensures that the password meets the various requirements set in the password policy.

When you establish a password policy, you set the following types of rules, to mention just a few:

- The maximum length of time a given password is valid
- The minimum number of characters a password must contain
- The number of numeric characters required in a password

*See Also:* Chapter 19, "Password Policies in Oracle Internet Directory" for a fuller description of the rules you set when establishing password policies
Authentication by Using Simple Authentication and Security Layer (SASL)

The section "Direct Authentication" on page 16-4 introduced the use of SASL within an Oracle Internet Directory environment. This section describes more fully how SASL works. It contains these topics:

- How a SASL-Enabled Client Authenticates to a Directory Server by Using Digest-MD5
- How a SASL-Enabled Client Authenticates to a Directory Server by Using External Authentication

How a SASL-Enabled Client Authenticates to a Directory Server by Using Digest-MD5

When a SASL-enabled client seeks Digest-MD5 authentication to a server, the authentication process is as follows:

1. The directory server sends to the LDAP client a digest-challenge that includes various Digest-MD5 authentication options that it supports and a special token.
2. The client selects an authentication option, then sends a digest-response to the server indicating the option it has selected. The response includes some secure tokens and a client credential in encrypted format. This allows it to authenticate itself to the server.
3. The directory server then decrypts and verifies the client credential from the response.

How a SASL-Enabled Client Authenticates to a Directory Server by Using External Authentication

Oracle Internet Directory provides SASL-external authentication over an SSL connection in which both client and server authenticate themselves to each other by providing certificates. The DN is derived from the client certificate used in the SSL network negotiation.

When a client seeks authentication to a directory server by using an external authentication mechanism such as SSL, the authentication process is as follows:

1. The client sends an initial message with the authorization identity.
2. The directory server uses information external to SASL to determine whether the client can validly authenticate as the authorization identity. If the client can validly authenticate, then the directory server indicates successful completion of the authentication exchange. Otherwise, the directory server indicates failure.

The system providing the external information may be IPsec or SSL/TLS. The authorization identity is derived as follows:

- In case of exact match, the authorization identity is derived from the client authentication credentials in the system providing external authentication—for example, the client SSL certificate.
- If the client sends an empty string as the authorization identity, then the authorization identity is derived from the client authentication credentials in the system providing external authentication—for example, the SSL certificate.
This chapter explains how to configure Secure Sockets Layer (SSL) for use with Oracle Internet Directory. If you use Secure Sockets Layer (SSL), you may also configure strong authentication, data integrity, and data privacy.

This chapter contains these topics:
- Supported Cipher Suites
- SSL Client Scenarios
- Limitations of the Use of SSL in 10g (10.1.4.0.1)
- Configuring and Testing Oracle Internet Directory With SSL
- Other Components and SSL

**See Also:** "Security" on page 3-14 for a conceptual overview of SSL in relation to Oracle Internet Directory

### Supported Cipher Suites

A cipher suite is 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.

Table 17–1 lists the SSL cipher suites supported by Oracle Internet Directory and their corresponding authentication, encryption, and data integrity mechanisms.

<table>
<thead>
<tr>
<th>Cipher Suite</th>
<th>Authentication</th>
<th>Encryption</th>
<th>Data Integrity</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSL_RSA_WITH_3DES_EDE_CBC_SHA</td>
<td>RSA</td>
<td>DES40</td>
<td>SHA</td>
</tr>
<tr>
<td>SSL_RSA_WITH_RC4_128_SHA</td>
<td>RSA</td>
<td>RC4_40</td>
<td>SHA</td>
</tr>
<tr>
<td>SSL_RSA_WITH_RC4_128_MD5</td>
<td>RSA</td>
<td>None</td>
<td>MD5</td>
</tr>
<tr>
<td>SSL_RSA_WITH_DES_CBC_SHA</td>
<td>RSA</td>
<td>None</td>
<td>SHA</td>
</tr>
<tr>
<td>SSL_DH_anon_WITH_3DES_EDE_CBC_SHA</td>
<td>-</td>
<td>3DES_EDE_CBC</td>
<td>SHA</td>
</tr>
<tr>
<td>SSL_DH_anon_WITH_RC4_128_MD5</td>
<td>-</td>
<td>RC4_40</td>
<td>MD5</td>
</tr>
<tr>
<td>SSL_DH_anon_WITH_DES_CBC_SHA</td>
<td>-</td>
<td>DES_CBC</td>
<td>SHA</td>
</tr>
<tr>
<td>SSL_RSA_EXPORT_WITH_RC4_40_MD5</td>
<td>-</td>
<td>RC4_40</td>
<td>MD5</td>
</tr>
</tbody>
</table>
SSL Client Scenarios

Oracle Internet Directory clients can use SSL 2.0 or SSL 3.0. A client over SSL can connect to a server anonymously or by using either simple or strong authentication. When both a client and server authenticate themselves to each other, SSL derives the identity information it requires from the X509v3 digital certificates.

Limitations of the Use of SSL in 10g (10.1.4.0.1)

In Oracle Internet Directory 10g (10.1.4.0.1), the Oracle directory replication server cannot communicate directly with an SSL-enabled LDAP server that supports two way (mutual) authentication. The replication server startup will fail and hang if the LDAP server is configured for SSL mutual authentication.

See Also: Chapter 7, "Oracle Directory Server Administration" for instructions on how to configure server instances.

Configuring and Testing Oracle Internet Directory With SSL

Use Oracle Wallet Manager to configure Oracle Internet Directory for SSL. To test the connection, use either the command line or Oracle Directory Manager.

Note: By default, the SSL port that is defined in configuration set 0 is set to authentication mode 1 (encryption only). Do not configure the SSL port of configset 0 with a authentication mode other than 1. Doing so will break Oracle Delegated Administration Services and other applications that expect to communicate with Oracle Internet Directory on the encrypted SSL port.

This section contains these topics:

- Configuring SSL Parameters
- Configure Oracle Internet Directory for SSL
- Testing SSL Connections From the Command Line
- Testing SSL Connections With Oracle Directory Manager

Configuring SSL Parameters

During start-up of a directory server instance, the directory reads a set of configuration parameters, including the parameters for the SSL profile. If you are going to run the directory with SSL enabled, you need to examine—and possibly reconfigure—the SSL parameters in the configuration set entry.

Table 17-1 (Cont.) SSL Cipher Suites Supported in Oracle Internet Directory

<table>
<thead>
<tr>
<th>Cipher Suite</th>
<th>Authentication</th>
<th>Encryption</th>
<th>Data Integrity</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSL_RSA_EXPORT_WITH_DES40_CBC_SHA</td>
<td>-</td>
<td>DES40</td>
<td>SHA</td>
</tr>
<tr>
<td>SSL_DH_anon_EXPORT_WITH_RC4_40_MD5</td>
<td>-</td>
<td>RC4_40</td>
<td>MD5</td>
</tr>
<tr>
<td>SSL_DH_anon_EXPORT_WITH_DES40_CBC_SHA</td>
<td>-</td>
<td>DES40</td>
<td>SHA</td>
</tr>
</tbody>
</table>
To run a server instance in secure mode, set the SSL Enable parameter in the configuration settings to 1: the default secure port is 3031. To allow the same instance to run non-secure connections concurrently, set SSL Enable to 2: the default non-secure port is 3060.

You can create and modify multiple sets of configuration parameters with differing values, using a different configuration set entry for each instance of Oracle Internet Directory. This is a useful way to accommodate clients with different security needs.

Oracle Corporation recommends that you create separate configuration sets and modify their SSL values, rather than modify SSL values in the default configuration set. The default set may be required by Oracle Support Services in the diagnosis of certain technical issues.

This section contains these topics:

- Configuring SSL Parameters by Using Oracle Directory Manager
- Configuring SSL Parameters by Using Command-Line Tools

### Configuring SSL Parameters by Using Oracle Directory Manager

You can examine and modify the values for the SSL configuration parameters in each configuration set entry that you have created and in each server instance that is currently running.

**Note:** You cannot directly change the parameters for an active instance. If you want to change the parameters for an active instance, change the parameters in a configuration set entry and save it. After it is saved, you can stop current instances and refer to the newly modified configuration set in the start server message.

### Viewing and Modifying SSL Configuration Parameters

**To view and modify SSL configuration parameters:**

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

2. Expand either Directory Server or Replication Server, as appropriate. The numbered configuration sets are listed beneath your selection.

3. Select the configuration set that you want to examine. The group of tab pages for that configuration set entry appear in the right pane.

4. Select the SSL Settings tab page, modify the fields and save the changes. These fields are described in Table A-47 on page A-32.

**See Also:** "Managing Server Configuration Set Entries by Using Oracle Directory Manager" on page 7-3 for information about changing parameters in a configuration set entry.
Configuring SSL Parameters by Using Command-Line Tools
For information about configuring SSL parameters from the command line, see:

- "Managing Server Configuration Set Entries by Using Command-Line Tools" on page 7-5
- "Oracle Internet Directory Server Administration Tools" in Oracle Identity Management User Reference for instructions on using the -p, -u, and -w flags to \ldapadd and related commands to configure SSL

Configure Oracle Internet Directory for SSL
Configure the server side LDAP server for SSL as follows:

1. Start the Oracle Wallet Manager
   On Unix, set the DISPLAY environment variable and type:
   \owm
   On Windows, start the program by using either:
   - Start, then Programs, then ORACLE_HOME, then Network Administration, then Wallet Manager
   - Start, then Programs, then ORACLE_HOME, then Integrated Management Tools, then Wallet Manager

2. Select Wallet from the top menu bar and then New.
   Choose and confirm the password.

3. A new empty wallet has been created.
   Select YES to create a certificate request.

4. Enter the required information.

5. Choose OK.
   An Oracle Wallet Manager dialog box informs you that a certificate request was successfully created. You can either copy the certificate request text from the body of this dialog panel and paste it into an e-mail message to send to a certificate authority, or you can export the certificate request to a file.

6. Choose Operations, then Export Certificate Request from the menu bar.
   The Export Certificate Request dialog box appears.

7. Enter a file name for the request, such as usercert.req.

8. Save the wallet.

   **Note:** When saving the wallet on Windows 2000, choose a directory path that does not contain spaces. Do not store the wallet in the default location, Documents and Settings\oracle\wallets.

9. Send the newly-created certificate request to your certificate authority.
10. You should receive a user certificate and, if needed, a trusted certificate from your certificate authority. If your CA is not in Oracle Wallet Manager's default list, you must import a trusted certificate from your CA before you can import the user certificate.

   a. To import the trusted certificate, choose Operations, then Import Trusted Certificate from the menu bar. The Import Trusted Certificate dialog panel appears. Choose either to paste the certificate in base64 format or to select a file containing the trusted certificate. Your new CA will appear in the list of Trusted Certificates.

   b. To import the user certificate, choose Operations, then Import Trusted Certificate from the menu bar. The Import Certificate dialog box appears. Choose either to paste the certificate in base64 format or to select a file containing the trusted certificate.

11. Select Wallet and save the wallet by selecting Wallet, then Save. Enable Auto Login by choosing Wallet from the menu bar, then choosing the check box next to the Auto Login menu item. A message at the bottom of the window displays Auto Login Enabled. A file called cwallet.sso is now present in your wallet directory.

   Note: As of Oracle Internet Directory Release 9.0.2, only wallets in encrypted format, such as cwallet.sso, are supported. For that reason, you must use Oracle Wallet Manager to open the wallet and to enable Auto Login before you start an SSL instance.

12. Open the Oracle Directory Manager and choose to add a new Configuration Set. Do not modify the Default Configuration Set.

   Select the SSL Setup tab and enter the location of the wallet. For UNIX, the URL format is:

   file://path/directory_of_wallet

   For example:

   file://etc/ORACLE/WALLET

   For Windows, the URL format is:

   file:\device:\path\wallet_directory

   For example:

   file:d:\wallet

   Choose the SSL authentication method and configure the SSL port. The authentication methods are:
Choose the port for the SSL instance for the release.

**See Also:** Chapter 6, "Process Control of Oracle Internet Directory Components"

13. You now have three configuration sets: `DefaultConfigset` with a default SSL port and a default non-SSL port, `defaultConfigset1` and your `newConfigset2` with a unique SSL port and unique non-SSL port.

On Windows systems, you must perform an extra configuration step. You must change the login account of the Oracle Directory Service from a local system account to the account of the user who owns the wallet. This user must be member of `Administrator Group`. Change the account as follows:

a. On Windows, choose **Start**, then **Settings**, then **Control Panel**, then **Administrative Tools**, then **Services**.

b. Click **PROPERTIES/LOGON**.

c. Change from **Local System Account** to the account you logged in as when you created the Wallet. Stop and restart the service.

14. Start the Oracle Internet Directory instances so that Oracle Delegated Administration Services and other applications requiring SSL in encrypted mode can operate normally.

a. Open a browser to the Oracle Enterprise Manager Web site and drill down into the Oracle Internet Directory processes. This page shows the running processes.

b. Click the Button **Start New Instance**. The new configuration set will be listed.

c. Select the **Set Number** to be started and click **Start**.

d. After the instance is started, click **OK** and the Oracle Internet Directory instances page will be displayed. The new instance will be shown in the list as started.

From this point on, the standard commands

```
opmnctl startall
opmnctl stopall
```

will automatically manage the Oracle Internet Directory instances.

15. You now have Oracle Internet Directory running and listening on four ports.

On a UNIX system, you can run the `$ORACLE_HOME/ldap/bin/ldapcheck` command to view the additional `oidldapd` dispatcher and server processes. The
debugging logs for the SSL instance are oidldapd02.log and oidldapd02sXXXXX.log, respectively.

Testing SSL Connections From the Command Line

You can use the ldapbind command to test SSL connections. On UNIX, the syntax is:

```
ldapbind -D cn=orcladmin -w welcome -U authentication_mode -h host -p SSL_port \
-W "file://DIRECTORY_CONTAINING_WALLET" -P wallet_password
```

and on Windows, the syntax is:

```
ldapbind -D cn=orcladmin -w welcome -U authentication_mode -h host -p SSL_port \
-W "file:device:\DIRECTORY_CONTAINING_WALLET" -P wallet_password
```

where authentication_mode is one of:

<table>
<thead>
<tr>
<th>Number</th>
<th>Authentication</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>No SSL authentication required.</td>
</tr>
<tr>
<td>2</td>
<td>One-way (server only) SSL authentication required.</td>
</tr>
<tr>
<td>3</td>
<td>Two-way (client and server) SSL authentication required.</td>
</tr>
</tbody>
</table>

See Also: The ldapbind command-line tool reference in Oracle Identity Management User Reference.

Testing SSL With Encryption Only

Use this method to test an SSL configuration with no SSL authentication required. The syntax is:

```
ldapbind -D cn=orcladmin -w password -U 1 -h host -p SSL_Port
```

Testing SSL With Server Authentication

Use this method to test an SSL configuration with SSL server authentication configured. A client can request either server authentication or no authentication.

For an anonymous bind with server authentication, the syntax is:

```
ldapbind -U 2 -h host -p port -W "file:DIRECTORY_CONTAINING_WALLET" \
-P wallet_password
```

For a bind with user "cn=orcladmin" and server authentication, the syntax is:

```
ldapbind -D cn=orcladmin -w password -U 2 -h host -p port \
-W "file:DIRECTORY_CONTAINING_WALLET" -P wallet_password
```

For a bind without SSL authentication, the syntax is:

```
ldapbind -D cn=orcladmin -w password -U 1 -h host -p SSL_Port
```

Testing SSL With Client and Server Authentication

Use this method to test an SSL configuration with SSL client and server authentication configured.

As of Oracle Internet Directory 10g (10.1.4.0.1), Oracle Internet Directory supports the Certificate Matching Rule. The DN and password passed on the ldapbind command
line are ignored. Only the DN from the certificate or the certificate hash is used for authorization.

**See Also:** "Direct Authentication" on page 16-4.

To bind with user "cn=orcladmin", the syntax is:

```
ldapbind -D cn=orcladmin -w password> -U 3 -p port \
-W "file:DIRECTORY_CONTAINING_WALLET" -P wallet_password
```

or

```
ldapbind -D cn=orcladmin -w password -U 2 -h host -p port \
-W "file:DIRECTORY_CONTAINING_WALLET" -P wallet_password
```

To use the bind DN (Distinguished Name) from the client certificate, the syntax is:

```
ldapbind -U 3 -h host -p port -W "file:DIRECTORY_CONTAINING_WALLET" \
-P wallet_password
```

or

```
ldapbind -U 2 -h host -p port -W "file:DIRECTORY_CONTAINING_WALLET" \
-P wallet_password
```

**Testing SSL Connections With Oracle Directory Manager**

To test the SSL connection with the Oracle Directory Manager, perform the following steps:

   
   At the login screen, click the Network Icon and add the new SSL instance.
   
   Choose the hostname and the port number of your configured SSL instance.

2. It should show AVAILABLE. Highlight it and click SELECT.

3. Click the SSL tab and fill in the wallet location of the user and password. For Windows, specify the SSL location as
   
   `file:device:\wallet_directory_path`

   For UNIX, specify the SSL Location as
   
   `file://wallet_directory_path`

   For SSL Password, specify your wallet password.

   For SSL Authentication Level, specify your configured authentication level.

4. Click the Credentials tab. Make sure the SSL check box is checked. If you omit this step, Oracle Directory Manager might hang.

5. Specify values for User and Password.

**Other Components and SSL**

At installation, Oracle Internet Directory starts up with configset0, which specifies dual mode. That is, some components can access Oracle Internet Directory using non-SSL connections, while others use SSL when connecting to the directory. By default, Oracle Application Server components are configured to run in this dual
Secure Sockets Layer (SSL) and the Directory

Secure Sockets Layer (SSL) and the Directory environment when communicating with Oracle Internet Directory. If you wish, you can remove the non-SSL mode and change all middle-tier instances to use SSL. For more information, please refer to the section on changing Oracle Internet Directory from dual mode to SSL mode in *Oracle Application Server Administrator’s Guide*.

Enterprise User Security or a customer application might need an SSL channel with a different configuration from configset0. For example, it might need SSL server authentication mode or SSL mutual authentication mode. In this case, you must configure an additional SSL mode port another configuration set so that an additional Oracle Internet Directory LDAP instance listens at that port.

---

**Note:** You should never modify the SSL mode of configset0. The modification might conflict with the default configuration of some Oracle Application Server components. You should use a different configuration set for a new SSL setup.

---

For more information about Enterprise User Security SSL configuration, please see the section on enterprise user security configuration in *Oracle Database Enterprise User Administrator’s Guide*.

Examples:

1. A configuration set for SSL server authentication mode:

   ```
   cn=configset2, cn=osidapd, cn=subconfigsubentry
   cn=configset2
   objectclass=top
   objectclass=orclConfigSet
   objectclass=orclLDAPSubConfig
   orclsslauthentication=32
   orclsslenable=2
   orclsslwalleturl=file:/ade/qdinh_newld/oracle/work/ldap/lrgsrg
   orclsslport=6060
   orclnonsslport=8019
   orclserverprocs=1
   ```

2. A configuration set for SSL mutual authentication mode:

   ```
   cn=configset3, cn=osidapd, cn=subconfigsubentry
   cn=configset3
   objectclass=top
   objectclass=orclConfigSet
   objectclass=orclLDAPSubConfig
   orclsslauthentication=64
   orclsslenable=2
   orclsslwalleturl=file:/ade/qdinh_newld/oracle/work/ldap/lrgsrg
   orclsslport=7001
   orclnonsslport=8029
   orclserverprocs=1
   ```
This chapter provides an overview of access control policies and describes how to administer directory access control by using either Oracle Directory Manager or the command-line tool, ldapmodify.

**Note:** As of Oracle Internet Directory 10g (10.1.4.0.1), the super user is subject to access control policies like any other user. The new ACL syntax changes for super user restriction cannot be administered through Oracle Directory Manager.

The access control policies described in this chapter allow the delegation of coarse-grained privileges across an entire realm. If you need finer-grained delegation, you might wish to use Oracle Access Manager.

**See Also:** Oracle Access Manager Access Administration Guide

This chapter contains these topics:
- Overview of Access Control Policy Administration
- How ACL Evaluation Works
- Managing Access Control by Using Oracle Directory Manager
- Managing Access Control by Using Command-Line Tools

**See Also:**
- "Security" on page 3-14 and Chapter 16, "Directory Security Concepts" for a conceptual explanation before you begin implementing and administering access control policies
- Appendix C, "The Access Control Directive Format" for information about the format or syntax of Access Control Items (ACIs)

### Overview of Access Control Policy Administration

You manage access control policies by configuring the values of the ACI attributes within appropriate entries. You can do this by using either Oracle Directory Manager or ldapmodify.

This section contains these topics:
Access Control Management Constructs

This section discusses the structures used for access control in Oracle Internet Directory. These include:

- Access Control Policy Points (ACPs)
- The `orclACI` attribute for prescriptive access control
- The `orclEntryLevelACI` attribute for entry-level access control
- Privilege Groups

Access Control Policy Points (ACPs)

ACPs are entries in which the `orclACI` attribute has been given a value. The `orclACI` attribute value represents the access policies that are inherited by the subtree of entries starting with the ACP as the root of the subtree.

When a hierarchy of multiple ACPs exists in a directory subtree, a subordinate entry in that subtree inherits the access policies from all of the superior ACPs. The resulting policy is an aggregation of the policies within the ACP hierarchy above the entry.

For example, if an ACP is established in the HR department entry, and the Benefits, Payroll, and Insurance groups are entries within the HR department, then any entry within those groups inherits the access rights specified in the HR department entry.

When there are conflicting policies within a hierarchy of ACPs, the directory applies well-defined precedence rules in evaluating the aggregate policy.

See Also:  "How ACL Evaluation Works" on page 18-10

The `orclACI` Attribute for Prescriptive Access Control

The `orclACI` attribute contains access control list (ACL) directives that are prescriptive—that is, these directives apply to all entries in the subtree below the ACP where this attribute is defined. Any entry in the directory can contain values for this attribute. Access to this attribute itself is controlled in the same way as access to any other attribute.

Note:  It is possible to represent ACL directives specific to a single entry in the `orclACI` attribute. However, in such scenarios, for administrative convenience and performance advantages, Oracle Corporation recommends using `orclEntryLevelACI`—discussed in "The `orclEntryLevelACI` Attribute for Entry-Level Access Control". This is because the LDAP operational overhead increases with the number of directives represented through `orclACI`. You can reduce this overhead by moving entry specific directives from `orclACI` to `orclEntryLevelACI`.

The `orclEntryLevelACI` Attribute for Entry-Level Access Control

When a policy pertains only to a specific entity—for example, a special user—you can maintain the ACL directives within the entry for that entity. You do this by using a
user-modifiable operational attribute called orclEntryLevelACI. This attribute contains ACL directives only for the entry with which it is associated.

Any directory entry can optionally carry a value for this attribute. This is because Oracle Internet Directory extends the abstract object class top to include orclEntryLevelACI as an optional attribute.

The orclEntryLevelACI attribute is multi-valued and has a structure similar to that of orclACI.

**See Also:** "Object: To What Are You Granting Access?" on page 18-6 for the structure definition of the orclEntryLevelACI attribute

**Security Groups**

Group entries in Oracle Internet Directory are associated with either the groupOfNames or the groupOfUniqueNames object class. Membership in the group is specified as a value of the member or uniqueMember attribute respectively.

To specify access rights for a group of people or entities, you identify them in security groups. There are two types of security groups: ACP groups and privilege groups.

**ACP groups** If an individual is a member of an ACP group, then the directory server simply grants to that individual the privileges associated with that ACP group.

Use ACP groups to resolve access at the level of an ACP. For example, suppose you want to give to several hundred users access to browse an entry. You could assign the browse privilege to each entry individually, but this could require considerable administrative overhead. Moreover, if you later decide to change that privilege, you would have to modify each entry individually. A more efficient solution is to assign the privilege collectively. To do this, you create a group entry, designate it as an ACP group, assign the desired privilege to that group, then assign users as members of that group. If you later change the access rights, you need to do it in one place, for the group, rather than for each individual user. Similarly, you can remove that privilege from multiple users by removing them from the group, rather than having to access multiple individual entries.

ACP groups are associated with the orclacpgroup object class.

**Privilege Groups** A privilege group is a higher-level access group. It is similar to an ACP group in that it lists users with similar rights. However, it also provides for additional checking beyond a single ACP, as follows: if an ACP denies access, an attribute in the user's entry tells the directory server whether the user being denied is in any privilege group. If so, then this user has additional rights at a higher administration level, and all higher administration levels in the DIT are checked. If the directory server finds a higher ACP that grants to the privilege group access to the requested object, then it overrides the denials by the subordinate ACP, and grants access to the user. If, however, the orclACI or orclEntryLevelACI attribute of a subordinate ACP contains the keyword DenyGroupOverride, the higher level ACP does not override the subordinate ACP. Use DenyGroupOverride to restrict super user access through privileged groups.

Normally, you would implement only ACP groups. The additional checking that privilege groups provide can degrade performance. Use privilege groups only when access control at higher levels needs the right to override standard controls at lower levels.
Use privilege groups to grant access to administrators who are not recognized by ACPs lower in the DIT. For example, suppose that the global administrator in a hosted environment must perform operations in a realm. Because the global administrator’s identity is not recognized in the realm of the hosted company, the directory server, relying only on the ACPs in that realm, denies the necessary access. However, if the global administrator is a member of a privilege group, then the directory server looks higher in the DIT for an ACP that grants to this privilege group the access rights to that subtree. If it finds such an ACP, then the directory server overrides the denials by ACPs in the hosted company’s realm.

Add the `DenyGroupOverride` keyword to an ACI to deny access to members of privileged groups.

Privilege groups are associated with the `orclPrivilegeGroup` object class.

**Users in Both Types of Groups** If a user is a member of both an ACP group and a privilege group, then the directory server performs an evaluation for each type of group. It resolves access rights for the privilege group by looking to ACPs higher in the DIT.

**Overview: Granting Access Rights to a Group** To grant access rights to a group of users, you do the following:

1. Create a group entry in the usual way.
2. Associate the group entry with either the `orclPrivilegeGroup` object class or the `orclACPgroup` object class.
3. Specify the access policies for that group.
4. Assign members to the group.

**How the Directory Server Computes Security Group Membership** Entries can have either direct memberships in groups, or indirect memberships in other ACP or privilege groups by means of nested groups, thus forming a forest of privilege groups. Access policies specified at a given level are applicable to all the members directly or indirectly below that level.

Because Oracle Internet Directory evaluates for access control purposes only security groups, it does not allow setting access policies for other types of groups. When a user binds with a specific distinguished name (DN), Oracle Internet Directory computes the user’s direct membership in security groups. Once it knows the first level groups for the given DN, Oracle Internet Directory computes nesting of all these first level groups into other security groups. This process continues until there are no more nested groups to be evaluated.

Each security group, nested or otherwise, must be associated with a security group object class—either `orclACPgroup` or `orclPrivilegeGroup`. Even if a group is a member of a security group, the directory server does not consider it for access control purposes unless it is associated with a security group object class. When it has determined the user’s membership in security groups, the directory server uses that information for the lifetime of the session.

**Example: Computing Security Group Membership** For example, consider the sample group of entries in Table 18-1, each of which, with the exception of Group 4, is marked as a privilege group (`objectclass:orclPrivilegeGroup`). You can set access control policies that apply to the members of group1, group2, and group3.
Table 18–1  Sample Security Groups

<table>
<thead>
<tr>
<th>Group</th>
<th>Entry</th>
</tr>
</thead>
</table>
| Group 1 | dn: cn=group1,c=us  
          cn: group1  
          objectclass: top  
          objectclass: groupofUniqueNames  
          objectclass: orclPrivilegeGroup  
          uniquemember: cn=mary smith,c=us  
          uniquemember: cn=bill smith,c=us  
          uniquemember: cn=john smith,c=us |
| Group 2 | dn: cn=group2,c=us  
          cn: group2  
          objectclass: top  
          objectclass: groupofUniqueNames  
          objectclass: orclPrivilegeGroup  
          uniquemember: cn=mary jones,c=us  
          uniquemember: cn=joe jones,c=us  
          uniquemember: cn=bill jones,c=us  
          uniquemember: cn=john smith,c=us |
| Group 3 | dn:cn=group3,c=us  
          cn: group3  
          objectclass: top  
          objectclass: groupofUniqueNames  
          objectclass: orclPrivilegeGroup  
          uniquemember: cn=group2,c=us  
          uniquemember: cn=group1,c=us  
          uniquemember: cn=group4,c=us |
| Group 4 | dn: cn=group4,c=us  
          cn: group4  
          objectclass: top  
          objectclass: groupofUniqueNames  
          uniquemember: cn=john doe,c=uk  
          uniquemember: cn=jane doe,c=uk  
          uniquemember: cn=anne smith,c=us |

Group 3 contains the following nested groups:

- cn=group2,c=us
- cn=group1,c=us
- cn=group4,c=us

Access control policies for Group 3 are applicable to members of Group 3, Group 1, and Group 2 because each of them is marked as a privilege group. These same access control policies are not applicable to the members of Group 4 because Group 4 is not marked as a privilege group.

For example, suppose that the user binds to Oracle Internet Directory as a member of Group 4 with the DN cn=john doe,c=uk. None of the access policies applicable to the members of Group 3 will apply to this user. This is because his only direct membership is to a non-privilege group. By contrast, if the user were to bind as cn=john smith,c=us—that is, as a member of Group 1 and Group 2—then his access rights will be governed by access policies set up for members of Group 1, Group 2, as well as Group 3 (in which Group 1 and Group 2 are nested). This is because all three groups are associated with the object class orclPrivilegeGroup.
Access Control Information Components

Access control information represents the permissions that various entities or subjects have to perform operations on a given object in the directory. Thus, an ACI consists of three components:

- The object to which you are granting access
- The entities or subjects to whom you are granting access
- The kind of access you are granting

Object: To What Are You Granting Access?

The object part of the access control directive determines the entries and attributes to which the access control applies. It can be either an entry or an attribute.

Entry objects associated with an ACI are implicitly identified by the entry or the subtree where the ACI itself is defined. Any further qualification of objects at the level of attributes is specified explicitly in the ACL expressions.

In the orclACI attribute, the entry DN component of the object of the ACI is implicitly that of all entries within the subtree starting with the ACP as its topmost entry. For example, if dc=com is an ACP, then the directory area governed by its ACI is:

```
.*, dc=com.
```

However, since the directory area is implicit, the DN component is neither required nor syntactically allowed.

In the orclEntryLevelACI attribute, the entry DN component of the object of the ACL is implicitly that of the entry itself. For example, if dc=acme,dc=com has an entry level ACI associated with it, then the entry governed by its ACI is exactly:
 dc=acme,dc=com. Since it is implicit, the DN component is neither required nor syntactically allowed.

The object portion of the ACL allows entries to be optionally qualified by a filter matching some attribute(s) in the entry:

```
filter=(ldapFilter)
```

where ldapFilter is a string representation of an LDAP search filter. The special entry selector * is used to specify all entries.

Attributes within an entry are included in a policy by including a comma-delimited list of attribute names in the object selector.

```
attr=(attribute_list)
```

Attributes within an entry are excluded from a policy by including a comma-delimited list of attribute names in the object selector.

```
attr!=(attribute_list)
```

The object part of an access control directive may also include special keywords. These are:
DenyGroupOverride, which prevents access from being overridden by higher level ACPs

AppendToAll, which causes the subject of an ACI to be added to all other ACIs in that ACP during evaluation.

---

**Note:** Access to the entry itself must be granted or denied by using the special object keyword ENTRY. Note that giving access to an attribute is not enough; access to the entry itself through the ENTRY keyword is necessary.

---

**See Also:** Appendix C, "The Access Control Directive Format" for information about the format or syntax of ACIs

**Subject: To Whom Are You Granting Access?**

This section describes:

- The entity being granted access
- The bind mode—that is, the authentication mode used to verify the identity of that entity
- The added object constraint, which limits the kind of objects a user can add below the parent once access is granted.

**Entity** Access is granted to entities, not entries. The entity component identifies the entity or entities being granted access.

You can specify entities either directly or indirectly.

**Directly specifying an entity:** This method involves entering the actual value of the entity, for example group=managers. You can do this by using:

- The wildcard character (*), which matches any entry
- The keyword `SELF`, which matches the entry protected by the access
- The keyword `SuperUser`, which matches the `SuperUser` DN specified in the directory.
- A regular expression, which matches an entry's distinguished name, for example, `dn=regex`
- The members of a privilege group object: `group=dn`

**Indirectly specifying an entity:** This is a dynamic way of specifying entities. It involves specifying a DN-valued attribute that is part of the entry to which you are granting access. There are three types of DN-valued attributes:

- `dnattr`: Use this attribute to contain the DN of the entity to which you are granting or denying access for this entry.
- `groupattr`: Use this attribute to contain the DNs of the administrative groups to which you are granting or denying access for this entry.
- `guidattr`: Use this attribute to contain the global user identifier (orclGUID) of the entry to which you want to grant or deny access for this entry.

For example, suppose you want to specify that Anne Smith's manager can modify the salary attribute in her entry. Instead of specifying the manager DN directly, you...
specify the DN-valued attribute: \texttt{dnattr=manager}. Then, when John Doe seeks to modify Anne’s salary attribute, the directory server:

1. Looks up the value for her \texttt{manager} attribute and finds it to be John Doe
2. Verifies that the bind DN matches the \texttt{manager} attribute
3. Assigns to John Doe the appropriate access

\textbf{Bind Mode} The bind mode specifies the methods of authentication and of encryption to be used by the subject.

There are four authentication modes:

- MD5Digest
- PKCS12
- Proxy
- Simple: Simple password-based authentication

There are three encryption options:

- SASL
- SSL No Authentication
- SSL One Way

Specifying the encryption mode is optional. If it is not specified, then no encryption is used, unless the selected authentication mode is PKCS12. Data transmitted by using PKCS12 is always encrypted.

There is a precedence rule among authentication choices, and it is as follows:

Anonymous < Proxy < Simple < MD5Digest < PKCS12

This rule means that:

- Proxy authentication blocks anonymous access
- Simple authentication blocks both Proxy and Anonymous access
- MD5Digest authentication blocks Simple, Proxy and Anonymous access
- PKCS12 authentication blocks MD5Digest, Simple, Proxy and Anonymous access

The bind mode syntax is:

\[
\text{BINDMODE} = (\text{LDAP\_AUTHENTICATION\_CHOICE} + [ \text{LDAP\_ENCRYPTION\_CHOICE} ]) \\
\text{LDAP\_AUTHENTICATION\_CHOICE} = \text{Proxy} | \text{Simple} | \text{MD5Digest} | \text{PKCS12} \\
\text{LDAP\_ENCRYPTION\_CHOICE} = \text{SSL\_No\_Auth} | \text{SSL\_One\_Way} | \text{SASL}
\]

The \texttt{LDAP\_ENCRYPTION\_CHOICE} is an optional parameter. If you do not specify it, then the directory server assumes that no encryption is to be used.

\textbf{Added Object Constraint} When a parent entry has \texttt{add} access, it can add objects as entries lower in the hierarchy. The added object constraint can be used to limit that right by specifying an \texttt{ldapfilter}.

\textbf{See Also:} Appendix C, "The Access Control Directive Format" and Appendix B, "The LDAP Filter Definition"

\textbf{Operations: What Access Are You Granting?} The kind of access granted can be one of the following:
- None
- Compare/nocompare
- Search/nosearch
- Read/noread
- Selfwrite/noselfwrite
- Write/nowrite
- Add/noadd
- Proxy/noproxy
- Browse/nobrowse
- Delete/nodelete

Note that each access level can be independently granted or denied. The noxxx means xxx permission is denied.

Note also that some access permissions are associated with entries and others with attributes.

<table>
<thead>
<tr>
<th>Access Level</th>
<th>Description</th>
<th>Type of Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compare</td>
<td>Right to perform compare operation on the attribute value</td>
<td>Attributes</td>
</tr>
<tr>
<td>Read</td>
<td>Right to read attribute values. Even if read permission is available for an attribute, it cannot be returned unless there is browse permission on the entry itself.</td>
<td>Attributes</td>
</tr>
<tr>
<td>Search</td>
<td>Right to use an attribute in a search filter</td>
<td>Attributes</td>
</tr>
<tr>
<td>Selfwrite</td>
<td>Right to add yourself to, delete yourself from, or modify your own entry in a list of DNs group entry attribute. Use this to allow members to maintain themselves on lists. For example, the following command allows people within a group to add or remove only their own DN from the member attribute: access to attr=(member) by dnattr=(member) (selfwrite) The dnattr selector indicates that the access applies to entities listed in the member attribute. The selfwrite access selector indicates that such members can add or delete only their own DN from the attribute.</td>
<td>Attributes</td>
</tr>
<tr>
<td>Write</td>
<td>Right to modify/add/delete the attributes of an entry.</td>
<td>Attributes</td>
</tr>
<tr>
<td>None</td>
<td>No access rights. The effect of granting no access rights to a subject-object pair is to make the directory appear to the subject as though the object were not present in the directory.</td>
<td>Both entries and attributes</td>
</tr>
<tr>
<td>Add</td>
<td>Right to add entries under a target directory entry</td>
<td>Entries</td>
</tr>
<tr>
<td>Proxy</td>
<td>Allows the subject to impersonate another user</td>
<td>Entries</td>
</tr>
<tr>
<td>Browse</td>
<td>Permission to return the DNs in the search result. It is equivalent to the list permission in X.500. This permission is also required for a client to use an entry DN as the base DN in an ldapsearch operation.</td>
<td>Entries</td>
</tr>
<tr>
<td>Delete</td>
<td>Right to delete the target entry</td>
<td>Entries</td>
</tr>
</tbody>
</table>

The entry level access directives are distinguished by the keyword ENTRY in the object component.
How ACL Evaluation Works

When a user tries to perform an operation on a given object, the directory server determines whether that user has the appropriate access to perform that operation on that object. If the object is an entry, it evaluates the access systematically for the entry and each of its attributes.

Evaluating access to an object—including an attribute of an entry—can involve examining all the ACI directives for that object. This is because of the hierarchical nature of ACPs and the inheritance of policies from superior ACPs to subordinate ACPs.

The directory server first examines the ACI directives in the entry-level ACI, `orclEntryLevelACI`. It proceeds to the nearest ACP, then considers each superior ACP in succession until the evaluation is complete.

During ACL evaluation, an attribute is said to be in one of the states described in Table 18–4:

<table>
<thead>
<tr>
<th>State</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resolved with permission</td>
<td>The required access for the attribute has been granted in the ACI.</td>
</tr>
<tr>
<td>Resolved with denial</td>
<td>The required access for the attribute has been explicitly denied in the ACI.</td>
</tr>
</tbody>
</table>

### Table 18–3 LDAP Operations and Access Needed to Perform Each One

<table>
<thead>
<tr>
<th>Operation</th>
<th>Required Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create an object</td>
<td>Add access to the parent entry</td>
</tr>
<tr>
<td>Modify</td>
<td>Write access to the attributes that are being modified</td>
</tr>
<tr>
<td>ModifyDN</td>
<td>Delete access to the current parent and Add access to the new parent</td>
</tr>
<tr>
<td>ModifyDN (RDN)</td>
<td>Write access to the naming attribute—that is, the RDN attribute</td>
</tr>
<tr>
<td>Remove an object</td>
<td>Delete access to the object being removed</td>
</tr>
<tr>
<td>Compare</td>
<td>Compare access to the attribute and Browse access to the entry</td>
</tr>
<tr>
<td>Search</td>
<td>• Search access on the filter attributes and Browse access on the entry (if only the entry DN needs to be returned as a result)</td>
</tr>
<tr>
<td></td>
<td>• Search access on the filter attributes, Browse access on the entry, and Read permission on the attributes (for all attributes whose values need to be returned as a result)</td>
</tr>
</tbody>
</table>

### Table 18–4 Attribute States During ACL Evaluation
In all operations except search, the evaluation stops if:

- Access to the entry itself is denied
- Any of the attributes reach the resolved with denial state

In this case the operation would fail and the directory server would return an error to the client.

In a search operation, the evaluation continues until all the attributes reach the resolved state. Attributes that are resolved with denial are not returned.

This section contains these topics:

- Precedence Rules Used in ACL Evaluation
- Use of More Than One ACI for the Same Object
- Exclusionary Access to Directory Objects
- ACL Evaluation For Groups

### Precedence Rules Used in ACL Evaluation

An LDAP operation requires the BindDN, or subject, of the LDAP session to have certain permissions to perform operations on the objects—including the entry itself and the individual attributes of the entry.

Typically, there could be a hierarchy of access control administration authorities, starting from the root of a naming context down to successive administrative points (or access control policy points). An ACP is any entry which has a defined value for the orclACI attribute. Additionally, the access information specific to a single entry can also be represented within the entry itself (orclEntryLevelACI).

ACL evaluation involves determining whether a subject has sufficient permissions to perform an LDAP operation. Typically an orclEntryLevelACI or orclACI might not contain all the necessary information for ACL evaluation. Hence, all available ACL information is processed in a certain order until the evaluation is fully resolved.

That order of processing follows these rules:

- The entry level ACI is examined first. ACIs in the orclACI are examined starting with the ACP closest to the target entry and then its superior ACP and so on.
- At any point, if all the necessary permissions have been determined, the evaluation stops; otherwise, the evaluation continues.
- Within a single ACI, if the entity associated with the session DN matches more than one item identified in the by clause, the effective access evaluates to:
  - The union of all the granted permissions in the matching by clause items ANDed with
  - The union of all the denied permissions in the matching by clause items

### Precedence at the Entry Level

ACIs at the entry level are evaluated in the following order:
How ACL Evaluation Works

1. With a filter. For example:
   access to entry filter=(cn=p*)
   by group1 (browse, add, delete)

2. Without a filter. For example:
   access to entry
   by group1 (browse, add, delete)

Precedence at the Attribute Level
At the attribute level, specified ACIs have precedence over unspecified ACIs.

1. ACIs for specified attributes are evaluated in the following order:
   a. Those with a filter. For example:
      access to attr=(salary) filter=(salary > 10000)
      by group1 (read)
   b. Those without a filter. For example:
      access to attr=(salary)
      by group1 (search, read)

2. ACIs for unspecified attributes are evaluated in the following order:
   a. With a filter. For example:
      access to attr=(*) filter (cn=p*)
      by group1 (read, write)
   b. Without a filter. For example:
      access to attr=(*)
      by group1 (read, write)

Use of More Than One ACI for the Same Object
Oracle Internet Directory, enables you to define more than one ACI in the ACP of an object. It processes the ACIs associated with that object and stores them as a single ACI in its internal ACP cache. It then applies all the relevant policies in the multiple ACIs specified in the ACP.

The following example of an ACP illustrates how this works.

Access to entry by dn="cn=john" (browse, noadd, nodelete)
Access to entry by group="cn=admingroup" (browse, add, nodelete)
Access to entry by dn=".*,c=us" (browse, noadd, nodelete)

In this ACP, there are three ACIs for the object entry. When it loads this ACP, Oracle Internet Directory merges these three ACIs as one ACI in its internal ACP cache.

The ACI syntax is:

Access to OBJECT> by SUBJECT ACCESSLIST
OBJECT = [ entry | attr [EQ-OR-NEQ] ( * | ATTRLIST ) ]
[ filter = ( LDAPFILTER ) ]

This syntax makes possible the following types of objects:

- Entry
- Entry + filter = ( LDAPFILTER )
How ACL Evaluation Works

- Attr = (ATTRLIST)
- Attr = (ATTRLIST) + filter = (LDAPFILTER)
- Attr != (ATTRLIST)
- Attr != (ATTRLIST) + filter = (LDAPFILTER)
- Attr = (*)
- Attr = (*) + filter = (LDAPFILTER)

You can define multiple ACIs for any of the above types of objects. During initial loading of the ACP, the directory server merges the ACIs based on which of these object types are defined. The matching criterion is the exact string comparison of the object strings in the ACIs.

If one ACI specifies Attr= (ATTRLIST) and another Attr!= (ATTRLIST), then Attr= (*) must not be specified as an ACI in the entry. Also, if an ACI specifies Attr= (ATTRLIST), then, to specify the access rights to attributes not in ATTRLIST, Attr= (*) must be used and not Attr!= (ATTRLIST). Attr= (*) implies all attributes other than those specified in ATTRLIST.

**Note:** When you define multiple ACIs on the same attribute with the same filter, Oracle Internet Directory merges them to create a single ACI in the runtime structure.

When you define multiple ACIs on the same attribute with different filters, Oracle Internet Directory treats them as separate ACIs. In such cases, the precedence order is non-deterministic.

To prevent ambiguous behavior, if you define multiple ACIs with different filters against the same attribute, ensure that the filters yield non-overlapping sets of results.

---

Exclusionary Access to Directory Objects

If an ACI exists for a given object, you can specify access to all other objects except that one. You do this either by granting access to all the objects, or by denying access to the one object.

In the following example, access is granted to all attributes:

```
access to attr= (*)
by group2 (read)
```

In the following example, access is denied to the `userpassword` attribute:

```
access to attr!= (userpassword)
by group2 (read)
```

---

ACL Evaluation For Groups

If an operation on an attribute or the entry itself is explicitly denied at an ACP low in the DIT, then, typically, the ACL evaluation for that object is considered "Resolved with Denial." However, if the user of the session (bindDN) is a member of a group object, then the evaluation continues as if it is still unresolved. If permissions are granted to the user of the session at an ACP higher in the tree through a group subject selector, then such grants have precedence over any denials lower in the DIT.
This scenario is the only case in which an ACL policy at a higher level ACP has precedence over an ACP policy lower in the DIT.

Managing Access Control by Using Oracle Directory Manager

You can view and modify access control information within ACPs by using either Oracle Directory Manager or command-line tools. This section explains how to accomplish these tasks by using Oracle Directory Manager.

---

**Note:** Immediately after installing Oracle Internet Directory, be sure to reset the default security configuration as described in "Task 1: Reset the Default Security Configuration" on page 4-1

---

**Note:** As of Oracle Internet Directory 10g (10.1.4.0.1), the super user is subject to access control policies like any other user. The new ACL syntax changes for super user restriction cannot be administered through Oracle Directory Manager.

This section contains these topics:

- Configuring Oracle Directory Manager for Access Control Management
- Viewing an ACP by Using Oracle Directory Manager
- Adding an ACP by Using Oracle Directory Manager
- Adding an ACP by Using the ACP Creation Wizard of Oracle Directory Manager
- Modifying an ACP by Using Oracle Directory Manager
- Granting Entry-Level Access by Using Oracle Directory Manager
- Example: Managing ACPs by Using Oracle Directory Manager

**See Also:** "Oracle Identity Management Command-line Tool Reference" in Oracle Identity Management User Reference for a description of command-line tools

Configuring Oracle Directory Manager for Access Control Management

You can configure how Oracle Directory Manager displays ACPs, and how it performs searches for ACPs.

Configuring the Display of ACPs in Oracle Directory Manager

Oracle Directory Manager 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, you may want to display them only as the result of a search.

To configure the display of ACPs:

1. In the navigator pane, expand Oracle Internet Directory Servers and select the server you want to configure.
2. On the toolbar, click User Preferences. The User Preferences dialog box appears.
4. Select either:
Always display all ACPs
- Only display ACPs based on search request

5. Choose OK.
6. To effect your changes, restart Oracle Directory Manager.

### Configuring Searches for ACPs When Using Oracle Directory Manager

For ACP searches, Oracle Directory Manager enables you to specify:
- The root of the search
- The maximum number of entries retrieved
- The time limit of the search
- The search depth

To configure searches for ACP entries:

1. In the navigator pane, expand **Oracle Internet Directory Servers** and select the directory server instance.
2. On the toolbar, choose **User Preferences**. The User Preferences dialog box appears.
3. Select the **Configure Entry Management** tab.
4. In the field labeled **Maximum number of one-level subtree entries**, enter the number of entries you want ACP searches to retrieve.
5. In the **Search Time Limit** field, enter the maximum number of seconds for the duration of the search.
6. Choose OK. A notice window displays the message "You need to restart Oracle Directory Manager to view ACP Management Changes."
7. Choose OK for the Notice window.
8. To view the latest access control management entries, disconnect and immediately reconnect Oracle Directory Manager.

### Viewing an ACP by Using Oracle Directory Manager

If you configured Oracle Directory Manager always to display ACPs, as described in "Configuring the Display of ACPs in Oracle Directory Manager" on page 18-14, then you can locate and view an ACP as follows:

1. In the navigator pane, expand **Oracle Internet Directory Servers**, then **directory server instance**, then **Access Control Management**. All of the defined ACPs appear in the navigator pane below the Access Control Management node.

2. In the navigator pane, under **Access Control Management**, select an ACP to display its information in the right pane. The fields in the Access Control Management pane are described in Table A–3 on page A-4.

If you configured Oracle Directory Manager to display ACPs only as the result of a search, as described in "Configuring the Display of ACPs in Oracle Directory Manager" on page 18-14, then you can locate and view an ACP as follows:

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

2. Perform a search for the entry designated as an ACP. The search result appears in the **Distinguished Name** box in the lower half of the right pane.
3. In the Distinguished Name box, double-click the entry. The corresponding Entry dialog box appears.

4. To view subtree access controls for this ACP, select the Subtree Access tab.
   To view entry level access controls for this ACP, select the Local Access tab.

**Adding an ACP by Using Oracle Directory Manager**

ACPs are entries that contain prescriptive, that is, inheritable, access control information. This information affects the entry itself and all entries below it. You will most likely create ACPs to broadcast large-scale access control throughout a subtree.

Adding an ACP by using Oracle Directory Manager involves three tasks:

- **Task 1: Specify the Entry That Will Be the ACP**
- **Task 2: Configure Structural Access Items**—that is, ACIs that pertain to entries
- **Task 3: Configure Content Access Items**—that is, ACIs that pertain to attributes

**Task 1: Specify the Entry That Will Be the ACP**

1. If you configured Oracle Directory Manager always to display ACPs, as described in "Configuring the Display of ACPs in Oracle Directory Manager" on page 18-14, then begin as follows:
   a. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance.
   b. Select Access Control Management, and go to step 2.

If you configured Oracle Directory Manager to display ACPs only as the result of a search, as described in "Configuring the Display of ACPs in Oracle Directory Manager" on page 18-14, then begin as follows:

   a. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance, then Access Control Management.
   b. Select a node where you want the ACP to reside. If there are no ACPs yet configured, then you may select ACPs under "DSE Root".


3. In the Path to Entry field, enter the distinguished name (DN) of the entry that will be the ACP. You can alternatively find the DN by choosing Browse to the right of the Path to Entry field.

**Task 2: Configure Structural Access Items**

1. To define structural access items, that is, ACIs that pertain to entries, just below the Structural Access Items window, choose Create. The Structural Access Item dialog box appears. It has four tabs: Entry Filter, Added Object Filter, By Whom, and Access Rights.

2. In an ACP, the access rights defined apply to the entry and all its subentries unless other filters restrict access further.
   If you want all entries below the ACP to be governed by the ACP, then you do not need to enter anything on the Entry Filter tab page; simply proceed to the next step. Otherwise, perform this step.
   If appropriate, use the Entry Filters tab page to identify the entries to which you are specifying access.
You might restrict access to an entry based on one or more of that entry’s attributes. For example, you might choose to restrict access to all entries in which the title is manager and in which the organization unit is Americas.

To identify an entry to which you are specifying access:

a. From the menu at the left end of the search criteria bar, select an attribute type.
b. From the menu in the middle of the bar, select one of the filter options. These options are described in Table A–45 on page A-31.
c. In the text box at the right end of the search criteria bar, type the value for the attribute you selected.

3. Select the **Added Object Filter** tab page.

You can specify ACIs to restrict the kind of entries a user can add. For example, you can specify an ACI in the DSE root entry that allows users to add only entries with `objectclass=country`. The directory server then verifies that any new entry complies with the constraints in this filter.

To restrict the kind of entries a user can add:

a. From the menu at the left end of the search criteria bar, select an attribute type.
b. From the menu in the middle of the bar, select one of the filter options. These options are described in Table A–45 on page A-31.
c. In the text box at the right of the search criteria bar, type the value for the attribute you selected.

4. Select the **By Whom** tab page.

   a. From the **Authentication Choice** list, select the type of authentication to be used by the subject (that is, the entity that seeks access). The options are described in Table A–4 on page A-4.

   If you do not choose an authentication method, then any kind of authentication is accepted. The authentication method specified on one node should match the one specified on the node it is communicating with.

   From the **Encryption Choice** list, select the type of encryption to be used. The options are described in Table A–5 on page A-4.

   b. Specify the entity or entities to whom you are granting access. The options are described in Table A–6 on page A-4.

5. Select the **Access Rights** tab page.

   Specify the kinds of rights to be granted:

   - **Browse**: Allows the subject to see the entry
   - **Add**: Allows the subject to add other entries below this entry
   - **Delete**: Allows the subject to delete the entry
   - **Proxy**: Allows the subject to impersonate another user

6. Click **OK**.

**Task 3: Configure Content Access Items**

1. To define content access items, that is, ACIs that pertain to attributes, just below the **Content Access Items** window, choose **Create**. The Content Access Item dialog box appears. Each tab page contains items you can modify.
2. If you want all entries below the ACP to be governed by the ACP, then you do not need to enter anything on Entry Filter tab page; simply proceed to Step 3. Otherwise, perform this step.

In an ACP, the access rights apply to the entry and all its subentries unless other filters restrict access further. If appropriate, use the Entry Filters tab page to identify the entries to which you are specifying access.

You might restrict access to an entry based on one or more of that entry’s attributes. For example, you might choose to restrict access to all entries in which the title is manager and in which the organization unit is Americas.

To identify an entry to which you are specifying access:

a. From the menu at the left end of the search criteria bar, select an attribute type.

b. From the menu in the middle of the bar, select one of the filter options. These are described in Table A–45 on page A-31.

c. In the text box at the right end of the search criteria bar, type the value for the attribute you selected.

3. Select the By Whom tab page.

a. From the Authentication Choice list, select the type of authentication to be used by the subject (that is, the entity that seeks access). The options are described in Table A–4 on page A-4.

If you do not choose an authentication method, then any kind of authentication is accepted. The authentication method specified on one node should match the one specified on the node it is communicating with.

From the Encryption Choice list, select the type of encryption to be used. The options are described in Table A–5 on page A-4.

b. Specify the entity or entities to whom you are granting access. The options are described in Table A–6 on page A-4.

4. Select the Attribute tab page.

a. From the right menu, select the attribute to which you want to grant or deny access.

b. From the left menu, select the matching operation to be performed against the attribute. Choices are EQ (Equal (=)) and NEQ (Not Equal (!=)).

For example, if you select EQ and cn, then the access rights you grant apply to the cn attribute. If you select NEQ and cn, then the access rights you grant do not apply to the cn attribute.

5. Select the Access Rights tab page and specify the privileges. These are described in Table A–7 on page A-5.

6. Click OK to close this dialog box and return to the main Oracle Directory Manager dialog box.

Adding an ACP by Using the ACP Creation Wizard of Oracle Directory Manager

The ACP Creation Wizard guides you through the tasks involved in adding an ACP. These tasks are:

- Task 1: Specify the Entry That Will Be the ACP
- Task 2: Configure Structural Access Items by Using the ACP Creation Wizard
Task 1: Specify the Entry That Will Be the ACP

1. If you configured Oracle Directory Manager always to display ACPs, as described in "Configuring the Display of ACPs in Oracle Directory Manager" on page 18-14, then begin as follows:
   a. In the navigator pane, expand Oracle Internet Directory Servers and directory server instance.
   b. In the navigator pane, select Access Control Management, and go to step 2.

   If you configured Oracle Directory Manager to display ACPs only as the result of a search, as described in "Configuring the Display of ACPs in Oracle Directory Manager" on page 18-14, then begin as follows:
   a. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance, then Access Control Management.
   b. In the navigator pane, select a node where you want the ACP to reside. If there are no ACPs yet configured, you may select ACPs under "DSE Root".

2. On the toolbar, click Create. A New Access Control Point dialog box appears.

3. In the Path to Entry field, enter the distinguished name (DN) of the entry that will be the ACP. You can alternatively find the DN by looking in the navigator pane under Entry Management or by clicking Browse.

In an ACP, the access rights apply either to the entry and all its subentries or to a specific entry only. The next sections tell you how to configure an ACP for either option.

Task 2: Configure Structural Access Items by Using the ACP Creation Wizard

1. To define structural access items, that is, ACIs that pertain to entries, just below the Structural Access Items window, click Create via Wizard. The first Structural Access Item dialog box appears.

2. If you specify prescriptive structural access items, then all entries below the ACP are governed by that ACP. If you want prescriptive structural access items, then you do not need to enter anything on this first Structural Access Item dialog box. Alternatively, if you want to grant access to a specific entry, then, in this first Structural Access Item dialog box, do the following:
   a. From the menu at the left end of the search criteria bar, select an attribute type.
   b. From the menu in the middle of the bar, select one of the filter options. These are described in Table A–45 on page A-31.
   c. In the text box at the right end of the search criteria bar, type the value for the attribute you selected.
   d. Click Next. A second Structural Access Item dialog box prompts you to specify any ACIs to restrict the kind of entries a user can add.

3. You can specify ACIs to restrict the kind of entries a user can add. For example, you can specify an ACI in the DSE root entry that allows users to add only entries with objectclass=country. The directory server then verifies that any new entry complies with the constraints in this filter.

   To restrict the kind of entries a user can add:
   a. From the menu at the left end of the search criteria bar, select an attribute type.
b. From the menu in the middle of the bar, select one of the filter options. These are described in Table A–45 on page A-31.

c. In the text box at the right end of the search criteria bar, type the value for the attribute you selected.

d. Choose Next. The wizard prompts you to choose the authentication and encryption methods, and the subject to whom you are granting access.

4. Specifying the authentication method is optional. If you do not set an authentication method, then any kind of authentication is accepted. The authentication method specified on one node must match the bind mode specified on the node it is communicating with.

a. To specify the type of authentication: From the Authentication Choice list, select the type of authentication to be used by the subject (that is, the entity that seeks access). The options are described in Table A–4 on page A-4.

b. To specify the type of encryption: From the Encryption Choice list, choose an encryption method. The options are described in Table A–5 on page A-4.

c. Specify the entity or entities to whom you are granting access. Options are described in Table A–6 on page A-4.

d. Click Next. A Structural Access Item dialog box prompts you for access rights information.

5. Specify the kinds of rights that are granted:

   ■ **Browse**: Allows the subject to see the entry
   ■ **Add**: Allows the subject to add other entries below this entry
   ■ **Delete**: Allows the subject to delete the entry
   ■ **Proxy**: Allows impersonating an entity without providing its password

6. Click Finish.

**Task 3: Configure Content Access Items by Using the ACP Creation Wizard**

1. To define content access items, that is, ACI that pertain to attributes, just below the Content Access Items window, click Create via Wizard. The first Content Access Item dialog box appears.

2. If you specify prescriptive content access items, then all entries below the ACP are governed by that ACP. If you want prescriptive content access items, then you do not need to enter anything on this first Content Access Item dialog box.

   Alternatively, to identify an attribute to which you are specifying access:

   a. From the menu at the left end of the search criteria bar, select an attribute type.
   b. From the menu in the middle of the bar, select one of the filter options. These are described in Table A–33 on page A-20.
   c. In the text box at the right end of the search criteria bar, type the value for the attribute you selected.
   d. Click Next. A second Content Access Item dialog box prompts you to specify to whom you are granting access.
   e. Choose Next. The wizard prompts you to choose the authentication and encryption methods, and the subject to whom you are granting access.
3. Specifying the authentication method is optional. If you do not set an authentication method, then any kind of authentication is accepted. The authentication method specified on one node must match the bind mode specified on the node it is communicating with.
   
a. To specify the type of authentication, from the **Authentication Choice** list, select the type of authentication to be used by the subject (that is, the entity that seeks access). The options are described in Table A–4 on page A-4.
   
b. To specify the type of encryption, from the **Encryption Choice** list, choose an encryption method. The options are described in Table A–5 on page A-4.
   
c. Specify the entity or entities to whom you are granting access. Options are described in Table A–6 on page A-4.
   
d. Click Next. A Content Access Item dialog box prompts you to select an attribute and the matching operation to be performed against it.

4. To select an attribute and the matching operation to be performed against it:
   
a. In the Attribute field of the Content Access Item dialog box, from the right list, select the attribute to which you want to grant or deny access.
   
b. From the left list, select the matching operation to be performed against the attribute. Choices are EQ (Equal (=)) and NEQ (Not Equal (!=)).
   
c. Click Next. A Content Access Item dialog box prompts you to specify access rights.

5. Specify the kinds of rights to be granted. These are described in Table A–7 on page A-5.

6. Click Finish.

**Modifying an ACP by Using Oracle Directory Manager**

Modifying ACPs by using Oracle Directory Manager involves three tasks:

- Task 1: Specify the Entry That You Want to Modify.
- Task 2: Modify Structural Access Items—that is, ACIs that pertain to *entries*.
- Task 3: Modify Content Access Items—that is, ACIs that pertain to *attributes*.

**Task 1: Specify the Entry That You Want to Modify**

If you configured Oracle Directory Manager always to display ACPs, as described in "Configuring the Display of ACPs in Oracle Directory Manager" on page 18-14, then begin as follows:

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

2. Select **Access Control Management**. All of the defined Access Control Policy Points (ACPs) appear in a list below **Access Control Management** in the navigator pane. They also appear in the right pane.

3. In the navigator pane, under **Access Control Management**, select the ACP you want to modify. The information for that ACP is displayed in the right pane. Alternatively, you can double-click an ACP in the right pane to display the data in a separate dialog box.
If you configured Oracle Directory Manager to display ACPs only as the result of a search, as described in "Configuring the Display of ACPs in Oracle Directory Manager" on page 18-14, then begin as follows:

1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance, then Access Control Management.
2. Select the ACP you want to modify. The information for that ACP is displayed in the right pane.

**Task 2: Modify Structural Access Items**

You can add new structural access items, or modify existing ones.

**See Also:** "Task 2: Configure Structural Access Items" on page 18-16 for instructions about adding structural access items.

To modify structural access items:

1. In the Structural Access Items window, select the item you want to modify, and, just below the Structural Access Items window, click **Edit**. The Structural Access Item dialog box appears.
2. Use the **Entry Filters** tab page to narrow the set of entries to which you are granting access. If you want all entries below the ACP to be governed by the ACP, proceed to the next step.

You might choose an entry based on one or more attributes. For example, you might choose to search for all those whose title is secretary, or for all those whose title is manager and whose organization unit is Americas.

In the **Criteria** window of the **Entry Filters** tab page, use the search criteria bar to select an attribute, enter a value for that attribute, and specify a filter for matching the specified attribute with the value you entered. To do this:

   a. From the menu at the left end of the search criteria bar, select an attribute.
   b. From the menu in the middle of the bar, select one of the filter options. These are described in Table A–33 on page A-20.
   c. In the text box at the right end of the search criteria bar, type the value for the attribute you selected.

3. Use the **Added Object Filter** tab page to specify ACIs restricting the kind of entries a user can add. For example, you can specify an ACI in the DSE root entry that allows users to add only entries with objectclass=country. The directory server then verifies that any new entry complies with the constraints in this filter.

To restrict the kind of entries a user can add:

   a. From the menu at the left end of the search criteria bar, select an attribute type.
   b. From the menu in the middle of the bar, select one of the filter options. These are described in Table A–45 on page A-31.
   c. In the text box at the right end of the search criteria bar, type the value for the attribute you selected.

4. Use the **By Whom** tab page to specify the authentication and encryption methods, and the subject of the ACI (that is, the entity that seeks access).

Specifying the authentication method is optional. If you do not set an authentication method, then any kind of authentication is accepted. The
authentication method specified on one node must match the bind mode specified
on the node it is communicating with.

a. To specify the type of authentication: From the Authentication Choice list,
select the type of authentication to be used by the subject (that is, the entity
that seeks access). The options are described in Table A–4 on page A-4.

b. To specify the type of encryption: From the Encryption Choice list, choose an
encryption method. The options are described in Table A–5 on page A-4.

c. Specify the entity or entities to whom you are granting access. The options are
described in Table A–6 on page A-4.

5. Select the Access Rights tab page.

a. Determine what kinds of rights are granted:

   ■ Browse: Allows the subject to see the entry
   ■ Add: Allows the subject to add other entries below this entry
   ■ Delete: Allows the subject to delete the entry
   ■ Proxy: Allows impersonating an entity without providing its password

   If an entry is unspecified, then access is determined at the next highest level in
   which access is specified.

6. Click OK.

Task 3: Modify Content Access Items
You can add new content access items, or modify existing ones.

See Also: Task 3: Configure Content Access Items’ on page 18-17
for instructions about adding new content access items

To modify content access items:

1. In the Content Access Items box, select the content access item you want to
modify, then, just below the Content Access Items box, click Edit. The Content
Access Items dialog box appears. Each tab page contains items you can modify.

2. If you want all entries below the ACP to be governed by the ACP, then you do not
need to enter anything on Entry Filter tab page; simply proceed to the next step.

   In an ACP, the access rights defined apply to the entry and all its subentries unless
   other filters restrict access further. If appropriate, use the Entry Filters tab page to
   identify the entries to which you are specifying access.

   You might restrict access to an entry based on one or more of that entry’s
   attributes. For example, you might choose to restrict access to all entries in which
   the title is manager and in which the organization unit is Americas.

   To identify an entry to which you are specifying access:

   a. From the menu at the left end of the search criteria bar, select an attribute type.

   b. From the menu in the middle of the bar, select one of the filter options. These
   are described in Table A–45 on page A-31.

   c. In the text box at the right end of the search criteria bar, type the value for the
   attribute you selected.

3. Use the By Whom tab page to specify the authentication and encryption methods,
and the subject of the ACI (that is, the entity that seeks access).
Specifying the authentication method is optional. If you do not set an authentication method, then any kind of authentication is accepted. The authentication method specified on one node must match the bind mode specified on the node it is communicating with.

a. To specify the type of authentication, from the Authentication Choice list, select the type of authentication to be used by the subject (that is, the entity that seeks access). The options are described in Table A–4 on page A-4.

b. To specify the type of encryption, from the Encryption Choice list, choose an encryption method. The options are described in Table A–5 on page A-4.

c. Specify the entity or entities to whom you are granting access. The options are described in Table A–6 on page A-4.

4. Select the Attribute tab page.

a. From the right menu, select the attribute to which you want to grant or deny access.

b. From the left menu, select the matching operation to be performed against the attribute. Choices are EQ (Equal (=)) and NEQ (Not Equal (!=)).

For example, if you select EQ and cn, then the access rights you grant apply to the cn attribute. If you select NEQ and cn, then the access rights you grant do not apply to the cn attribute.

5. Select the Access Rights tab page and specify the privileges. These are described in Table A–7 on page A-5.

6. Click OK.

Granting Entry-Level Access by Using Oracle Directory Manager

To grant entry-level access by using Oracle Directory Manager:

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

2. In the navigator pane, select the entry to display its properties in the right pane.

3. Select the Local Access tab page, then create and edit local ACIs in the Structural Access Item and Content Access Item boxes as described in “Modifying an ACP by Using Oracle Directory Manager” on page 18-21.

4. Once you have made the changes, click Apply.

---

**Note:** You must click Apply to send the information you just entered to the directory server. Otherwise, the information is simply held in the Oracle Directory Manager cache.

Example: Managing ACPs by Using Oracle Directory Manager

This example illustrates how to use Oracle Directory Manager to create a new ACP that has ACIs within it. Suppose you are an administrator in a large company, and you want to limit access to user passwords, so that everyone can compare a password, but only the owner of each password, that is, the user, can read the password or modify it.

In this example, we create a new ACP and populate it with four ACIs that set the following permissions:

- Limited access to a userpassword attribute by everyone
Open access to the same userpassword attribute by the user himself
- Open access to all attributes except userpassword to everyone
- Open access to all attributes to everyone

Create a New ACP
1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance.
2. Select Access Control Management. A list of ACPs appears in the right pane.
3. At the bottom of the right pane, click Create. A New Access Control Point dialog box appears.
4. In the Path to Entry field, enter the DN where you want the ACP. The ACIs within the ACP will apply to all entries below and including that DN.

Configure Structural Access Items
To set the access rights for an entry:
   Because you want the ACIs to apply to all entries under the ACP, do not use the Entry Filter tab page.
2. Select the Added Object Filter tab page.
   You can specify ACIs to restrict the kind of entries a user can add. For example, you can specify an ACI in the DSE root entry that allows users to add only entries with objectclass=country. The directory server then verifies that any new entry complies with the constraints in this filter.
   To restrict the kind of entries a user can add:
   a. From the menu at the left end of the search criteria bar, select the objectclass attribute type.
   b. From the menu in the middle of the bar, select Exact Match.
   c. In the text box at the right of the search criteria bar, enter country.
   The Added Object Filter tab page should now look like Figure 18–1.
3. Select the **By Whom** tab page.
   a. From the **Authentication Choice** list, select **MD5Digest**.
   b. From the **Encryption Choice** list, choose **SASL**.
   c. To create access rights for everyone, select **Everyone**. The **By Whom** tab page should look like Figure 18–2.

![Figure 18–2 Structural Access Item: By Whom Tab Page](image)

4. Select the **Access Rights** tab page. By default, all rights—browse, add, and delete—are granted. Proxy is unspecified.
   a. Change the access rights so that everyone can browse all entries, but cannot add or delete them. The **Access Rights** tab page should look like Figure 18–3.
b. Click OK.

**Configure Content Access Items** The four ACIs in this example use the same structural access item information. They differ only in the content access they allow. The rest of this section describes how to create the content access for the ACIs.

To define the content access items:

1. Below the **Content Access Items** box, click **Create**. The Content Access Items dialog box appears.
   Because you want this ACI to apply to all entries under the ACP, do not use the **Entry Filter** tab page.
2. Select the **By Whom** tab page.
   a. From the **Authentication Choice** list, select **MD5Digest**.
   b. From the **Encryption Choice** list, choose **SASL**.
   c. To create access rights for everyone, select **Everyone**. The **By Whom** tab page should look like Figure 18–4.
3. Select the **Attribute** tab page. This page has two fields. The first has two choices: **EQ** (equals) and **NEQ** (not equals). The second sets the attribute.

Select **EQ** and select **userPassword**. The **Attribute** tab page should look like Figure 18–5.

4. Select the **Access Rights** tab page. By default, all permissions are granted. Change the permissions so that read, search, write, and compare are denied. The **Access Rights** tab page should look like Figure 18–6.
5. Click OK.
   You have completed one ACI.

Create Another ACI  Create another ACI that allows a user to read, write, search, and compare his own password.

1. Under the Content Access Items box, click Create. The Content Access Items dialog box appears.

2. Select the By Whom tab page.
   a. From the Authentication Choice list, select MD5Digest.
   b. From the Encryption Choice list, choose SASL.
   c. To create access rights for everyone, select When Session User's Distinguished Name (DN) Matches the Accessed Entry. The By Whom tab page should look like Figure 18–7.
3. Select the **Attribute** tab page. This tab page has two lists. The first has two choices: **EQ** (equals) and **NEQ** (not equals). The second sets the attribute.

   Select **EQ** and **userPassword**. The **Attribute** tab page should look like **Figure 18–8**.

4. Select the **Access Rights** tab page.

   Grant access to read, search, write, and compare. Leave selfwrite unspecified. The **Access Rights** tab page should look like **Figure 18–9**.
5. Click OK. You have now created two ACIs. One denies Everyone read, search, write, and compare access to the userPassword attribute. The second allows the owner of the password to read, search, write, and compare that attribute.

Create a Third ACI
The next ACI grants access to Everyone to read, search, and compare all attributes except userPassword. It denies write access.

1. Under the Content Access Items box, click Create to display the Content Access Items dialog box.

2. Select the By Whom tab page.
   a. From the Authentication Choice list, select MD5Digest.
   b. From the Encryption Choice list, choose SASL.
   c. To create access rights for everyone, select Everyone. The By Whom tab page should look like Figure 18–10.
3. Select the **Attribute** tab page.

   Select **NEQ** and **userPassword**.

   This combination means that any attribute that is *not* equal to **userpassword** is the object of the permissions in this ACI. The **Attribute** tab page should look like Figure 18–11.

4. Select the **Access Rights** tab page.
Grant access to read, search, and compare. Deny write access. Leave selfwrite unspecified. The **Access Rights** tab page should look like Figure 18–12.

*Figure 18–12  Content Access Item: Access Rights Tab Page*

5. Click **OK** to apply these permissions and close the dialog box.

**Create a Fourth ACI**

The next ACI grants access to Self to read, browse, and write all attributes except **userpassword**. Including this ACI avoids any ambiguity about whether Self has the same access permissions as Everyone to attributes other than **userPassword**.

1. Under the **Content Access Items** box, click **Create** to display the Content Access Items dialog box.

2. Select the **By Whom** tab page.
   a. From the **Authentication Choice** list, select **MD5Digest**.
   b. From the **Encryption Choice** list, choose **SASL**.
   c. To create access rights for everyone, select **When Session User's Distinguished Name (DN) Matches the Accessed Entry**. The By Whom tab page should look like Figure 18–13.
3. Select the **Attribute** tab page.

   From the lists, select **NEQ** and **userPassword**. This combination means that any attribute that is *not* equal to **userPassword** is the object of the permissions in this ACI. The **Attribute** tab page should look like Figure 18–14.

![Figure 18–14  Content Access Item: Attribute Tab Page](image)

4. Select the **Access Rights** tab page.

   Grant access to read, search, and write. Leave selfwrite unspecified. The **Access Rights** tab page should look like Figure 18–15.
Managing Access Control by Using Command-Line Tools

As described in "Overview of Access Control Policy Administration" on page 18-1, directory access control policy information is represented as user-modifiable operational attributes. You can manage it by using command-line tools, including ldapmodify and ldapmodifymt, to set and alter the values of these attributes.

To directly edit the ACI, you should understand the format and semantics of the directory representation of the ACI as described in Appendix C, "The Access Control Directive Format".

This section contains these topics:

- Example: Restricting the Kind of Entry a User Can Add
- Example: Setting Up an Inheritable ACP by Using ldapmodify
- Example: Setting Up Entry-Level ACIs by Using ldapmodify
- Example: Using Wild Cards
- Example: Selecting Entries by DN
- Example: Using Attribute and Subject Selectors
- Example: Granting Read-Only Access
- Example: Granting Selfwrite Access to Group Entries
- Example: Defining a Completely Autonomous Policy to Inhibit Overriding Policies

5. Click OK to apply these permissions and close the dialog box.
Managing Access Control by Using Command-Line Tools

Example: Restricting the Kind of Entry a User Can Add
You can specify ACIs to restrict the kind of entries a user can add. For example, you can specify an ACI in the DSE root entry that allows users to add only entries with objectclass=country. To do this, you use the added_object_constraint filter. The directory server then verifies that any new entry complies with the constraints in this filter.

The following example specifies that:

- The subject cn=admin,c=us can browse, add, and delete under organization entries.
- The subject cn=admin,c=us can add organizationalUnit objects under organization entries
- All others can browse under organization entries

access to entry filter=(objectclass=organization) by group="cn=admin,c=us" constraintonaddedobject=(objectclass=organisationalunit) (browse,add,delete) by * (browse)

Example: Setting Up an Inheritable ACP by Using ldapmodify
This example sets up subtree access permissions in an orclACI at the root DSE by using an LDIF file named my_ldif_file. Because this example refers to the orclACI attribute, this access directive governs all the entries in the DIT.

```bash
ldapmodify -v -h $1 -D "cn=Directory Manager, o=IMC, c=US" -w "controller" \ -f my_ldif_file
```

The LDIF file, my_ldif_file, contains the following:

dn:
changeType: modify
replace: orclaci
orclaci: access to entry
by dn="cn=directory manager, o=IMC, c=us" (browse, add, delete)
by * (browse, noadd, nodelete)
orclaci: access to attr=(*)
by dn="cn=directory manager, o=IMC, c=us" (search, read, write, compare)
by self (search, read, write, compare)
by * (search, read, nowrite, nocompare)

See Also:
- "LDIF File Formatting Rules and Examples" in Oracle Identity Management User Reference for information about how to format input by using LDIF, the required input format for line mode commands
- The ldapmodify command-line tool reference in Oracle Identity Management User Reference for information about how to run ldapmodify
- Appendix C, "The Access Control Directive Format" for information about the format or syntax of ACI
Example: Setting Up Entry-Level ACIs by Using ldifmodify

This example sets up entry-level access permissions in the orclEntryLevelACI attribute by using an LDIF file named my_ldif_file. Because this example refers to the orclentrylevelACI attribute, this access directive governs only the entry in which it resides.

ldapmodify -v -h myhost -D "cn=Directory Manager, o=IMC, c=US" -w "controller" \ 
-f my_ldif_file

The LDIF file, my_ldif_file, contains the following:

dn:
  changetype: modify
  replace: orclentrylevelaci
  orclentrylevelaci: access to entry
    by dn="cn=directory manager, o=IMC, c=us" (browse, add, delete)
    by * (browse, noadd, nodelete)
  orclentrylevelaci: access to attr=(*)
    by dn="cn=directory manager, o=IMC, c=us" (search, read, write, compare)
    by * (search, read, nowrite, nocompare)

Note: In this example, no DN value is specified. This means that this ACI pertains to the root DSE and its attributes only.

Example: Using Wild Cards

This example shows the use of wild cards (*) in the object and subject specifiers. For all entries within the acme.com domain, it grants to everyone browse permission on all entries, as well as read and search permissions on all attributes.

In the ACP at dc=com, the orclACI attribute is specified as follows:

access to entry by * (browse)
access to attr=(*) by * (search, read)

Note that, in order to enable reading the attributes, you must grant permission to browse the entries.

Example: Selecting Entries by DN

This example shows the use of a regular expression to select the entries by DN in two access directives. It grants to everyone read-only access to the address book attributes under dc=acme, dc=com access.

The orclACI attribute of dc=acme, dc=com is specified as follows:

access to entry by * (browse)
access to attr=(cn, telephone, email) by * (search, read)

The orclACI attribute of dc=us, dc=acme, dc=com is specified as follows:

access to entry by * (browse)
access to attr=(*) by dn="*.",dc=us,dc=acme,dc=com" (search, read)

Example: Using Attribute and Subject Selectors

This example shows the use of an attribute selector to grant access to a specific attribute, and various subject selectors. The example applies to entries in the
dc=us, dc=acme, dc=com subtree. The policy enforced by this ACI can be described as follows:

- For all entries within the subtree, the administrator has add, delete, and browse permissions. Others within the dc=us subtree can browse, but those outside it have no access to the subtree.
- The salary attribute can be modified by your manager and viewed by yourself. No one else has access to the salary attribute.
- The userPassword attribute can be viewed and modified by yourself and the administrator. Others can only compare this attribute.
- The homePhone attribute can be read and written by yourself and viewed by anyone else.
- For all other attributes, only the administrator can modify values. Everyone else can compare, search, read, but cannot update attribute values.

The orclACI attribute of dc=us, dc=acme, dc=com is specified as follows:

```
access to entry
by dn="cn=admin, dc=us,dc=acme,dc=com" (browse, add, delete)
by dn=".*", dc=us,dc=acme,dc=com" (browse)
by * (none)
```

```
access to attr=(salary)
by dnattr=(manager) (read, write)
by self (read)
by * (none)
```

```
access to attr=(userPassword)
by self (search, read, write)
by dn="cn=admin, dc=us,dc=acme,dc=com" (search, read, write)
by * (compare)
```

```
access to attr=(homePhone)
by self (search, read, write)
by * (read)
```

```
access to attr != (salary, userPassword, homePhone)
by dn="cn=admin, dc=us,dc=acme,dc=com" (compare, search, read, write)
by * (compare, search, read)
```

**Example: Granting Read-Only Access**

This example gives to everyone read-only access to address book attributes under dc=acme, dc=com. It also extends to everyone read access to all attributes within the dc=us, dc=acme, dc=com subtree only.

The orclACI attribute of dc=acme, dc=com is specified as follows:

```
access to entry by * (browse)
access to attr=(cn, telephone, email) by * (search, read)
```

The orclACI attribute of dc=us, dc=acme, dc=com is specified as follows:

```
access to entry by * (browse)
access to attr="*" by dn=".*", dc=us,dc=acme,dc=com" (search, read)
```
Example: Granting Selfwrite Access to Group Entries

This example enables people within the US domain to add or remove only their own name (DN) to or from the `member` attribute of a particular group entry—for example, a mailing list.

The `orclEntryLevelACI` attribute of the group entry is specified as follows:

- `access to attr=(member)`
- `by dn=".*, dc=us, dc=acme, dc=com" (selfwrite)`

Example: Defining a Completely Autonomous Policy to Inhibit Overriding Policies

This example denies group override. It uses the following DNs:

<table>
<thead>
<tr>
<th>Container</th>
<th>DN</th>
</tr>
</thead>
<tbody>
<tr>
<td>User container</td>
<td><code>cn=users, c=us</code></td>
</tr>
<tr>
<td>Sensitive data</td>
<td><code>cn=appdata</code></td>
</tr>
<tr>
<td>User admin group for this naming context</td>
<td><code>cn= user admin group, cn=users, c=us</code></td>
</tr>
<tr>
<td>Security admin group or this naming context</td>
<td><code>cn= security admin group, cn=users, c=us</code></td>
</tr>
<tr>
<td>Global password admin group for all naming contexts that reset passwords</td>
<td><code>cn=password admin group</code></td>
</tr>
</tbody>
</table>

The policy requirements for `c=us` are as follows:

- Users can browse and read their information.
- The user security admin can modify the information under `c=us` except for passwords and ACPs.
- The security admin group can modify policies under `c=us`.
- The global password admin and the user can reset a password.
- All other users have no permissions.
- This policy cannot be overridden.

Required ACP:

- Access to entry `DenyGroupOverride`
  - `by dn=".*, c=us"` (browse,noadd,nodelete)
  - `by group="cn=User admin group, cn=users,c=us"` (browse,add,delete)

- Access to `attr=(orclaci) DenyGroupOverride`
  - `by group="cn=security admin group, cn=users,c=us"` (search,read,write,compare)
  - `by * (none)`

- Access to `attr=(userpassword) DenyGroupOverride`
  - `by self (search,read,write,compare)`
  - `by group="cn=password admin group"` (search,read,write,compare)
  - `by * (none)`
Access to attr(*) DenyGroupOverride
by self (search, read, nowrite, compare)
by group="cn= User admin group, cn=users, c=us" (search, read, write, compare)
by * (none)
Password policies are sets of rules that govern how passwords are used. This chapter contains these topics:

- **About Password Policies**
- **Managing Password Policies, Accounts, and Passwords**
- **Password Policy Error Messages**

### About Password Policies

This section contains these topics:

- **What a Password Policy Is**
- **Fine-Grained Password Policies**
- **Default Password Policy**
- **Password Policy Attributes**
- **Directory Server Verification of Password Policy Information**

### What a Password Policy Is

Password policies are sets of rules that govern password syntax and how passwords are used. Password policies enforced by Oracle Internet Directory include:

- The maximum length of time a given password is valid
- The minimum number of characters a password must contain
- The minimum number of numeric characters required in a password
- The minimum number of alphabetic characters
- The minimum number of repeated characters
- The use of upper and lower case
- The minimum number of non-alphanumeric characters (that is, special characters)
- That users change their passwords periodically
- The minimum and maximum time between password changes
- The grace period for logins after password expiration, by time or by number of logins
That users cannot reuse previously used passwords

Fine-Grained Password Policies

In previous releases, Oracle Internet Directory supported only one password policy in each realm. As of Oracle Internet Directory 10g (10.1.4.0.1), Oracle Internet Directory supports multiple password policies in each realm. Another change in 10g (10.1.4.0.1) is that these policies can be applied to any subtree within that realm. This means that entry-specific password policies are now possible.

Password policies can be specified as being realm-specific or directory-wide in scope. To achieve the desired scope, you must create the password policy entry in the appropriate container. In Oracle Internet Directory 10g (10.1.4.0.1), password policies are populated under a "cn=pwdPolicies" container created under the "cn=common" entry in each realm. By default these containers contain a password policy with the RDN "cn=default". The directory specific default password policy, for example, will have the DN: cn=default, cn=pwdPolicies, cn=Common, cn=Products, cn=OracleContext.

Other policies can be created under the pwdPolicies container, with different RDNs. Figure 19–1 illustrates this scenario.

**Figure 19–1  Location of Password Policy Entries**

In 10g (10.1.4.0.1), unlike previous releases, password policies are completely decoupled from the orclcommonusersearchbase attribute in a realm-specific Common Entry. If you upgraded from an earlier release, during the upgrade the existing password policies would have been migrated to the new architecture. However, simply adding a DN to the orclcommonusersearchbase no longer guarantees that the realm's default password policy will be applied to the subtree rooted at that DN.

In Oracle Internet Directory 10g (10.1.4.0.1), once you define a password policy, you must perform a second step to apply the password policy to a subtree of the directory. You must populate the pwdPolicysubentry attribute with the DN of the desired password policy on an entry that is the root of a subtree the Administrator wants the policy to be applicable to. Figure 19–2 illustrates this. The pwdPolicysubentry at
l=us contains the DN of the default policy, "cn=default, cn=pwdPolicies, cn=Common, cn=Products, cn=OracleContext", so the default policy applies to the users in the US. The pwdPolicySubentry at l=uk contains the DN of the policy "cn=policy2, cn=pwdPolicies, cn=Common, cn=Products, cn=OracleContext", so policy2 applies to the users in the UK.

Figure 19–2 pwdPolicySubentry Attributes Populated with DN of Password Policy

At runtime, Oracle Internet Directory resolves the applicable password policy on an entry by looking for a populated pwdPolicySubentry attribute in the entry and applying the policy pointed to by its value. If a populated pwdPolicySubentry attribute does not exist, Oracle Internet Directory will traverse up the directory tree until the nearest ancestor entry with a populated pwdPolicySubentry is found and apply the password policy pointed to by its value.

In general, establishing a password policy requires doing the following:

1. Create a password policy entry in the appropriate container and associating it with the pwdPolicy object.
2. Create the desired policy by setting values for attributes defined under the pwdPolicy object class for the entry created in step 1.
3. Ensure that the orclpwdpolicyenable attribute is set to 1. If this is not set to 1, Oracle Internet Directory ignores the policy.

4. Add and populate a pwdPolicysubentry attribute with the policy's DN, at the root of the subtree being governed by that policy.

**See Also:** "Object Class Reference" in *Oracle Identity Management User Reference* for a list and descriptions of the attributes of the pwdPolicy object class, and those of the top object class that pertain to password policies

---

**Note:** Password policy entries for subtrees and users are replicated. Replicating the 10g (10.1.4.0.1) policies to a pre-10g (10.1.4.0.1) node will not adversely impact the functionality of that node. A pre-10g (10.1.4.0.1) node, however, cannot meaningfully interpret the 10g (10.1.4.0.1) password policies. It will continue to enforce the password policy in the realm Oracle context.

---

**Note:** You must protect password policy entries from anonymous access using Oracle Internet Directory's ACI infrastructure, described in Chapter 18, "Directory Access Control". This is particularly important when a password policy is weak, as that information can assist an attacker in compromising the directory.

---

**Default Password Policy**

The default password policy for Oracle Internet Directory enforces:

- Password expiration in 120 days
- Account lockout after 10 login failures. Except for the super user account, all accounts remain locked for a duration of 24 hours unless the passwords are reset by the directory administrator. A user account stays locked even after the lockout duration has passed unless the user binds with the correct password

If the super user account, `cn=orcladmin`, becomes locked, it stays locked until it is unlocked by using the OID Database Password utility. This utility prompts you for the ODS user password. After you enter the ODS password, it unlocks the account.

**See Also:**

The `oidpasswd` command-line tool reference in *Oracle Identity Management User Reference* for information on unlocking a super user account

*Troubleshooting Password Policies* on page L-7.

- A minimum password length of five characters with at least one numeric character
- Password expiry warning seven days prior to expiry
- Five grace logins allowed after password expiry

Beginning in Oracle Internet Directory, Release 9.0.4, the password policy entry in the Root Oracle Context applies to the super user, but only the password policy governing account lockout is enforced on that account.
About Password Policies

Password Policies in Oracle Internet Directory

19-5

The Oracle Internet Directory password policy is applicable to simple binds (based on the userpassword attribute), compare operations on the userpassword attribute, and SASL binds. It does not apply to SSL and proxy binds.

Password Policy Attributes

The following attributes affect password policy:

Table 19–1 Password Policy Attributes

<table>
<thead>
<tr>
<th>Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>pwdMinAge</td>
<td>The number of seconds that must elapse between user modifications to the password. The default is 0.</td>
</tr>
<tr>
<td>pwdMaxAge</td>
<td>The maximum time, in seconds, that a password can be valid. Upon reaching this age, the password is considered to have expired. The default is 10368000 seconds (120 days).</td>
</tr>
<tr>
<td>pwdLockout</td>
<td>When this is true, the server locks out a user after a number of consecutive invalid login attempts. The number is specified by pwdMaxFailure. The default value of pwdLockout is 1 (true).</td>
</tr>
<tr>
<td>orclpwdIPLockout</td>
<td>When this is true, the server locks out a user after a number of consecutive invalid login attempts from the same IP address. The number is specified by orclpwdIPMaxFailure. The default is false.</td>
</tr>
<tr>
<td>pwdLockoutDuration</td>
<td>The time period in seconds to lock out a user account once the threshold of invalid login attempts is reached. The default is 86400 seconds (24 hours).</td>
</tr>
</tbody>
</table>
### Table 19–1 (Cont.) Password Policy Attributes

<table>
<thead>
<tr>
<th>Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>orclpwdIPLockoutDuration</code></td>
<td>The time period in seconds to lock out a user account once the threshold of invalid login attempts from the same IP address is reached. The default is 0.</td>
</tr>
<tr>
<td><code>pwdMaxFailure</code></td>
<td>The maximum number of invalid login attempts the server should allow before locking out a user account. The default value is 10.</td>
</tr>
<tr>
<td><code>orclpwdIPMaxFailure</code></td>
<td>The maximum number of invalid login attempts the server should allow from a particular IP address before locking the user account. The default is 0.</td>
</tr>
<tr>
<td><code>pwdFailureCountInterval</code></td>
<td>The time in seconds after which the password failures are purged from the failure counter, even though no successful authentication occurred. The default is 0.</td>
</tr>
<tr>
<td><code>pwdExpireWarning</code></td>
<td>The maximum number of seconds before a password is due to expire that expiration warning messages will be returned to an authenticating user. The default value is 604800 seconds (seven days).</td>
</tr>
<tr>
<td><code>pwdCheckSyntax</code></td>
<td>Enables or disables password syntax check</td>
</tr>
<tr>
<td></td>
<td>0–Disable all syntax checks</td>
</tr>
<tr>
<td></td>
<td>1–Enable password syntax value checks, except for encrypted passwords (default)</td>
</tr>
<tr>
<td><code>pwdMinLength</code></td>
<td>The minimum length of a password governed by this policy. The default is 5 characters</td>
</tr>
<tr>
<td><code>pwdGraceLoginLimit</code></td>
<td>The maximum number of grace logins allowed after a password expires. The default is 5.</td>
</tr>
<tr>
<td><code>orclpwdGraceLoginTimeLimit</code></td>
<td>The maximum period of time in seconds where grace logins are allowed after a password expires. If <code>orclpwdGraceLoginTimeLimit</code> is nonzero, then <code>pwdGraceLoginLimit</code> must be zero. If <code>pwdGraceLoginLimit</code> is nonzero, then <code>orclpwdGraceLoginTimeLimit</code> must be zero (the default).</td>
</tr>
<tr>
<td><code>pwdMustChange</code></td>
<td>Requires users to reset their password upon their first login after account creation or after a password has been reset by the administrator. The default is 0 (false).</td>
</tr>
<tr>
<td><code>orclpwdIllegalValues</code></td>
<td>A list of values that are not allowed as passwords.</td>
</tr>
<tr>
<td><code>orclpwdAlphaNumeric</code></td>
<td>The minimum number of numeric characters required in a password. The default is 1.</td>
</tr>
<tr>
<td><code>orclpwdMinAlphaChars</code></td>
<td>The minimum number of alphabetic characters required in a password. The default is 0.</td>
</tr>
<tr>
<td><code>orclpwdMinSpecialChars</code></td>
<td>The minimum number of non-alphanumeric characters (that is, special characters) required in a password. The default is 0.</td>
</tr>
<tr>
<td><code>orclpwdMinUppercase</code></td>
<td>The minimum number of uppercase characters required in a password. The default is 0.</td>
</tr>
<tr>
<td><code>orclpwdMinLowercase</code></td>
<td>The minimum number of lowercase characters required in a password. The default is 0.</td>
</tr>
<tr>
<td><code>orclpwdMaxRptChars</code></td>
<td>The maximum number of repeated characters allowed in a password. The default is 0.</td>
</tr>
<tr>
<td><code>pwdInHistory</code></td>
<td>The maximum number of used passwords stored in the <code>pwdHistory</code> attribute of a given entry. Passwords stored in <code>pwdHistory</code> cannot be used as a new password until they are purged from it. The default is 0.</td>
</tr>
</tbody>
</table>
Directory Server Verification of Password Policy Information

As explained in "Fine-Grained Password Policies" on page 19-2, Oracle Internet Directory determines the applicable policy for an entry by locating the appropriate populated pwdPolicysubentry. To ensure that the user password meets the requirements of a given policy, the directory server verifies:

- That the password policy is enabled. It does this by checking the value of the attribute orclpwdpolicyenable in the password policy entry. A value of 1 indicates that the password policy is enabled. A value of 0 indicates that it is disabled.

- Correctness of password policy syntax information, which includes, for example, the correct number of alphabetic and numeric characters, or the correct password length. The directory server checks the syntax during ldapadd and ldapmodify operations on the userpassword attribute.

- Password policy state information, which, for example, includes:
  - The timestamp of the user password creation or modification
  - That the minimum password age is greater than the current time minus the time of password creation
  - The timestamp of consecutive failed login attempts by the user
  - The time at which the user account was locked
  - Indicator that the password has been reset and must be changed by the user on first authentication
  - A history of user's previously used passwords
  - Time stamps of grace logins

If the grace login is set by time period, the server checks the time discrepancy between the current time and the expiration.

The directory server checks the state information during ldapbind and ldapcompare operations, but does so only if the orclpwdpolicyenable attribute is set to 1.

To enable password value syntax checking, set the attributes orclpwdpolicyenable and pwdchecksyntax in the password policy entry to TRUE.
Managing Password Policies, Accounts, and Passwords

This section contains these topics:

- Managing Password Policies by Using Oracle Directory Manager
- Managing Accounts and Passwords by Using the Self-Service Console

Table 19–2 lists the administrative tasks related to password policies and the tools you use to perform each one, and points you to the corresponding information.

<table>
<thead>
<tr>
<th>Task</th>
<th>Tools</th>
<th>Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabling and disabling accounts</td>
<td>Oracle Internet Directory Self-Service Console, ldapmodify</td>
<td>&quot;Enabling and Disabling Accounts by Using the Oracle Internet Directory Self-Service Console&quot; on page 19-12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Example: Enabling and Disabling Accounts by Using Command-Line Tools&quot; on page 19-11</td>
</tr>
<tr>
<td>Forcing a password change</td>
<td>ldapmodify</td>
<td>&quot;Example: Forcing a Password Change by Using Command-Line Tools&quot; on page 19-12</td>
</tr>
<tr>
<td>Modifying password policies for an</td>
<td>Oracle Directory Manager, ldapmodify</td>
<td>&quot;Creating Password Policies by Using Oracle Directory Manager&quot; on page 19-9</td>
</tr>
<tr>
<td>Setting password policies</td>
<td>ldapmodify</td>
<td>&quot;Example: Setting Password Policies by Using Command-Line Tools&quot; on page 19-10</td>
</tr>
<tr>
<td>Unlocking accounts</td>
<td>Oracle Internet Directory Self-Service Console, ldapmodify</td>
<td>&quot;Unlocking Accounts by Using the Oracle Internet Directory Self-Service Console&quot; on page 19-12</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&quot;Example: Unlocking Accounts by Using Command-Line Tools&quot; on page 19-11</td>
</tr>
<tr>
<td>Viewing password policies for an</td>
<td>Oracle Directory Manager, ldapsearch</td>
<td>&quot;Viewing Password Policies by Using Oracle Directory Manager&quot; on page 19-9</td>
</tr>
<tr>
<td>identity management realm</td>
<td></td>
<td>&quot;Example: Viewing Password Policies by Using Command-Line Tools&quot; on page 19-10</td>
</tr>
</tbody>
</table>

Managing Password Policies by Using Oracle Directory Manager

You can use Oracle Directory Manager to view, refresh, and modify password policies. This section contains these topics:

- Viewing Password Policies by Using Oracle Directory Manager
- Modifying Password Policies by Using Oracle Directory Manager
Managing Password Policies, Accounts, and Passwords

Creating Password Policies by Using Oracle Directory Manager

Viewing Password Policies by Using Oracle Directory Manager
To view the password policies, in the navigator pane, expand Oracle Internet Directory Servers, then directory server instance, then Password Policy Management. The navigator pane displays the password policy entries. The right pane has two tabs. The General tab displays the path to password policy group entry. The Password Policy Effective Subtree tab displays a table with two columns:

- The Password Policy column listing each password policy entry
- The Effective Subtree column listing the subtree to which each policy applies

To get the latest updates to password policies, choose Refresh.

To get a specific password policy, in the navigator pane, choose the password policy you want to view. The policies appear in the right pane.

See Also: "Password Policy Fields in Oracle Directory Manager" on page A-6 for a description of each password policy displayed in Oracle Directory Manager

Modifying Password Policies by Using Oracle Directory Manager
To modify the password policies:

1. In the navigator pane, expand in succession Oracle Internet Directory Servers, directory server instance, Password Policy Management.
2. In the navigator pane, choose the password policy you want to modify. The corresponding tab pages appear in the right pane.
3. In the General tab page, modify the editable attribute fields as needed. These fields are described in Table A–10 on page A-6.
4. Select the Account Lockout tab page and, to modify the fields, select Global Lockout. Modify the editable attribute fields as needed. These fields are described in Table A–11 on page A-7.
5. Select the IP Lockout tab page and, to modify the fields, select IP Lockout. Modify the editable attribute fields as needed. These fields are described in Table A–12 on page A-7.
6. Select the Password Syntax tab page and, to modify the fields, select Check Password Syntax. Modify the editable attribute fields as needed. These fields are described in Table A–13 on page A-8.
7. Select the Effective Subtree tab page to modify the subtree to which the policy applies.
8. When you are finished, choose Apply.

Creating Password Policies by Using Oracle Directory Manager
To create a new password policy:

1. In the navigator pane, expand in succession Oracle Internet Directory Servers, directory server instance, Password Policy Management.
2. In the navigator pane, choose one of the existing password policies. The corresponding tab pages appear in the right pane.
3. In the right pane, select the name of the policy, then select Edit.
Managing Password Policies, Accounts, and Passwords

4. To create a new policy, select **Create** or **Create Like**.

5. In the **General** tab page, set or modify the editable attribute fields as needed. These fields are described in Table A–10 on page A-6.

6. Select the **Account Lockout** tab page and, to modify the fields, select **Global Lockout**. Modify the editable attribute fields as needed. These fields are described in Table A–11 on page A-7.

7. Select the **IP Lockout** tab page and, to modify the fields, select **IP Lockout**. Modify the editable attribute fields as needed. These fields are described in Table A–12 on page A-7.

8. Select the **Password Syntax** tab page and, to modify the fields, select **Check Password Syntax**. Modify the editable attribute fields as needed. These fields are described in Table A–13 on page A-8.

9. Select the **Effective Subtree** tab page, then select **Add**. Either enter the DN, or select **Browse**, then use the **Select Distinguished Name (DN) Path** window to navigate to the subtree to which you want the policy to apply.

10. When you are finished, choose **Apply**.

Managing Password Policies, Accounts, and Passwords by Using Command-Line Tools

This section contains these topics:

- Example: Setting Password Policies by Using Command-Line Tools
- Examples: Managing Password Policies by Using Command-Line Tools
- Example: Enabling and Disabling Accounts by Using Command-Line Tools
- Example: Unlocking Accounts by Using Command-Line Tools
- Example: Forcing a Password Change by Using Command-Line Tools

**Example: Setting Password Policies by Using Command-Line Tools**

The following example disables the `pwdLockout` attribute, changing it from its default setting of 1.

The file `my_file.ldif` contains:

```
dn: cn=default,cn=pwdPolicies,cn=common,cn=products,cn=OracleContext, 
   o=my_company,dc=com 
changetype:modify 
replace: pwdlockout 
pwdlockout: 0 
```

The following command loads this file into the directory:

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

**Examples: Managing Password Policies by Using Command-Line Tools**

Look at the following examples to learn how to view and modify the password policies of a realm by using command-line tools.

**Example: Viewing Password Policies by Using Command-Line Tools**

The following example retrieves a specific password policy entry.

```
ldapsearch -p port -h host 
   -b "cn=pwdPolicies,cn=common,cn=products,cn=OracleContext, 
```
Managing Password Policies, Accounts, and Passwords

Password Policies in Oracle Internet Directory

The following example retrieves all password policy entries:

```bash
ldapsearch -p port -h host -b " " -s sub "(objectclass=pwdpolicy)"
```

Example: Modifying Password Policies by Using Command-Line Tools
The following example modifies a password policy entry.

```bash
ldapmodify -p port -h host -w <<EOF
dn: cn=default,cn=pwdPolicies,cn=common,cn=products,cn=OracleContext,
o=my_company,dc=com
changetype: modify
replace: pwdMaxAge
pwdMaxAge: 10000
EOF
```

Example: Enabling and Disabling Accounts by Using Command-Line Tools
You can temporarily disable a user's account, then enable it once again, by using command-line tools.

To permanently disable the account by setting the orclisenabled attribute to DISABLED. Setting this attribute to any other value enables the account.

To enable the account after you have disabled it, delete this attribute from the entry.

To enable the account for a specific period, set the orclActiveStartDate and orclActiveEndDate attributes in the user entry to the proper value in **UTC (Coordinated Universal Time)** format. For example:

```bash
cn=John Doe,cn=users,o=my_company,dc=com
orclactivestartdate:20030101000000z
orclactiveenddate: 20031231000000z
```

In this example, John Doe can log in only between January 1, 2003 and December 31, 2003. He cannot login prior to January 1, 2003 or after December 31, 2003. If you want to disable his account for a period of time between these dates, then set the orclisenabled attribute to FALSE.

Example: Unlocking Accounts by Using Command-Line Tools
If you are a member of the Security Administrators Group, then you can unlock an account without resetting the user password. This saves you from having to explicitly tell the user the new password. The user can simply log in using the old password.

To unlock an account, set the orclpwdaccountunlock attribute to 1.

The following example unlocks the account for user John Doe.

```bash
ldapmodify -p port -h host -D cn=orcladmin -w welcome -v <<EOF
dn: cn=John Doe,cn=users,o=my_company,dc=com
changetype: modify
add: orclpwdaccountunlock
orclpwdaccountunlock: 1
EOF
```
Example: Forcing a Password Change by Using Command-Line Tools
You can force users to change their passwords when they log in for the first time. To do this, set the pwdMustChange attribute in the pwdpolicy entry to TRUE, and then reset the password. If you do this, you must explicitly tell the user the new password so that the user can login to change that password.

See Also: "Resetting Your Own Password by Using the Oracle Internet Directory Self-Service Console" on page 19-12 for instructions on resetting passwords

Managing Accounts and Passwords by Using the Self-Service Console
This section explains how to use the Oracle Internet Directory Self-Service Console to:

- Enable and disable accounts
- Unlock accounts
- Reset your own password

Enabling and Disabling Accounts by Using the Oracle Internet Directory Self-Service Console
You can temporarily disable a user's account, then enable it once again, by using the Oracle Internet Directory Self-Service Console.

See Also: The section on managing accounts in Oracle Identity Management Guide to Delegated Administration for instructions on enabling and disabling accounts by using the Oracle Internet Directory Self-Service Console

Unlocking Accounts by Using the Oracle Internet Directory Self-Service Console
If you are a member of the Security Administrators Group, then, if an account becomes locked, you can unlock it without resetting the user password. This saves you from having to explicitly tell the user the new password. The user can simply log in by using the old password.

See Also: The section on managing accounts in Oracle Identity Management Guide to Delegated Administration for instructions on using the Oracle Internet Directory Self-Service Console to unlock accounts

Resetting Your Own Password by Using the Oracle Internet Directory Self-Service Console
If you forget your password or become locked out of your account, then you can reset your password. This involves identifying yourself to the server by providing values for a set of password validation attributes. This takes the form of answering a password hint question to which you had earlier specified an answer.

See Also: The section on resetting your password if you forget it in Oracle Identity Management Guide to Delegated Administration for instructions on using the Oracle Internet Directory Self-Service Console to reset your password
Password Policy Error Messages

Whenever there are password policy violations, the directory server sends to the client various error and warning messages. In Oracle Internet Directory, 10g (10.1.4.0.1), the directory server can send these messages as LDAP controls only if the client sends a password policy request control as a part of an ldapbind or ldapcompare operation. If the client does not send the request control, then the directory server does not send the response controls. Instead, it sends errors and warnings as part of additional information.

See: "Troubleshooting Password Policies" on page L-7 for a list of the messages and information about how to resolve them.
Password verifiers are the security credentials used to authenticate users to Oracle components other than Oracle Internet Directory. This chapter explains how Oracle Internet Directory centrally stores these password verifiers.

This chapter contains these topics:

- About Centralized Storage of User Authentication Credentials
- Storing and Managing Password Verifiers for Authenticating to Oracle Internet Directory
- Storing and Managing Password Verifiers for Authenticating to Oracle Components
- Verifier Generation Using Dynamic Parameters

About Centralized Storage of User Authentication Credentials

When a user leaves a company or changes jobs, that user’s privileges should change the same day to guard against misuse of old or unused accounts and privileges. Without centralized password administration, an administrator in a large enterprise with user accounts and passwords distributed over many databases may not be able make the changes as quickly as good security requires.

Oracle Internet Directory centrally stores security credentials to make their administration easy for both end users and administrators. It stores:

- Passwords for authenticating users to the directory itself
- Password verifiers for authenticating users to other Oracle components

Users can store non-Oracle authentication credentials if the non-Oracle applications are directory enabled. These applications must create their own container under the Products entry.

Storing and Managing Password Verifiers for Authenticating to Oracle Internet Directory

Oracle Internet Directory stores a user’s directory password in the userPassword attribute. You can protect this password by storing it as a Base64 encoded string of a one-way hashed value by using one of Oracle Internet Directory’s supported hashing algorithms. Storing passwords as one-way hashed values—rather than as encrypted values—more fully secures them because a malicious user can neither read nor decrypt them.
The default userPassword hashing algorithm for Oracle Internet Directory has been changed from MD4 to SHA-1. This default scheme is in effect for new installations only. All userPassword attributes created after a new install will be one-way hashed using SHA-1, then stored in Oracle Internet Directory.

When you perform an upgrade, the default hashing scheme in effect prior to upgrade is retained. For example, if the default scheme prior to the upgrade was MD4, then MD4 remains the default scheme after the upgrade. To ensure greater security of userPasswords, change the default scheme to SHA-1 immediately after the upgrade. When you change the default scheme to SHA-1, user login is unaffected. For greater security, require users to reset their passwords so that SHA-1 values hash values are stored in Oracle Internet Directory.

Oracle Internet Directory stores the user password in a reversible encrypted format in an operational attribute called orclrevpwd. This attribute is generated only if the attribute orclpwdencryptionenable in the password policy entry is set to 1. The orclrevpwd attribute can be queried only by using the SSL one-way and two-way authentication mechanisms. This attribute cannot be queried over non-SSL sessions.

This section contains these topics:

- Password Verifiers and Authentication to the Directory
- Hashing Schemes for Creating Password Verifiers
- Managing Password Protection by Using Oracle Directory Manager
- Managing Password Protection by Using ldapmodify

Password Verifiers and Authentication to the Directory

During authentication to a directory server, clients supply a password to the directory server in clear text. The directory server hashes this password by using the hashing algorithm specified in the DSE attribute userpassword. It then verifies it against the hashed password stored in the binding entry's userPassword attribute. If the hashed password values match, then the server authenticates the user. If they do not match, then the server sends the user an "Invalid Credentials" error message.

For external users, Oracle Internet Directory generates the attribute orclrevpwd during authentication. In particular this attribute is generated when clients authenticate using ldapcompare on the user's cleartext password. If the attribute orclrevpwd does not exist, then the Oracle Internet Directory server generates this attribute using the cleartext password provided for authentication. However this attribute is not generated if external users are authenticated against Oracle Internet Directory using ldapbind.

Hashing Schemes for Creating Password Verifiers

During installation, Oracle Universal Installer prompts you to set the one-way hashing scheme for protecting user passwords to the directory. It presents you with these options:

- **MD4**: A one-way hash function that produces a 128-bit hash, or message digest
- **MD5**: An improved and more complex version of MD4
- **SHA**: Secure Hash Algorithm, which produces a 160-bit hash, longer than MD5. The algorithm is slightly slower than MD5, but the larger message digest makes it more secure against brute-force collision and inversion attacks.
■ SSHA: Salted Secure Hash Algorithm. This is similar to SHA, but is generated by using a random salt with the password.

■ SMD5: Salted MD5. This is similar to MD5, but is generated by using a random salt with the password.

■ UNIX Crypt: The UNIX hashing algorithm

The hashing algorithm value you specify at installation is stored in the orclCryptoScheme attribute in the root DSE. You can change that value by using either Oracle Directory Manager or ldapmodify.

Managing Password Protection by Using Oracle Directory Manager

You must be a super user to manage password protection by using Oracle Directory Manager.

To change the type of password protection by using Oracle Directory Manager:

1. In the navigator pane, expand Oracle Internet Directory Servers and select the directory server instance for which you want to reset password hashing. The corresponding tab pages for that directory server appear in the right pane.

2. In the System Operational Attributes tab page, in the Password Encryption field, select the type of password hashing you want to use. Options are:
   ■ MD4
   ■ MD5
   ■ No Encryption
   ■ SHA
   ■ UNIX Crypt
   ■ SSHA
   ■ SMD5

3. Choose Apply.

Note: The No Encryption option specifies that user passwords are stored in clear text.

Managing Password Protection by Using ldapmodify

The following example changes the password hashing algorithm to SHA by using an LDIF file named my_ldif_file:

ldapmodify -D cn=orcladmin -w welcome -h myhost -p 389 -v -f my_ldif_file

The LDIF file, my_ldif_file, contains:

dn: 
changetype: modify
replace: orclcryptoscheme
orclcryptoscheme: SHA

See Also: "Protection of User Passwords for Directory Authentication" on page 16-7
Storing and Managing Password Verifiers for Authenticating to Oracle Components

Oracle components store both passwords and password verifiers in Oracle Internet Directory. This section contains these topics:

- About Password Verifiers for Oracle Components
- Attributes for Storing Password Verifiers
- Default Verifiers for Oracle Components
- Example: How Password Verification Works for an Oracle Component
- Managing Password Verifier Profiles for Oracle Components by Using Oracle Directory Manager
- Managing Password Verifier Profiles for Oracle Components by Using Command-Line Tools

About Password Verifiers for Oracle Components

Oracle components can store their password values in Oracle Internet Directory as password verifiers. A password verifier is a hashed version of a clear text password, which is then encoded as a BASE64 encoded string.

You can choose one of these hashing algorithms to derive a password verifier:

- **MD5**: An improved, and more complex, version of MD4
- **SHA**: Secure Hash Algorithm, which produces a 160-bit hash, longer than MD5. The algorithm is slightly slower than MD5, but the larger message digest makes it more secure against brute-force collision and inversion attacks.
- **SSHA and SMD5**
- **UNIX Crypt**: The UNIX hashing algorithm
- **SASL/MD5**: Simple Authentication and Security Layer/MD5, which adds authentication support to connection-based protocols and uses a challenge-response protocol.
- **O3LOGON**: A proprietary Oracle algorithm for generating verifiers. It is similar to SASL/MD5 in that it uses a challenge-response protocol.
- **ORCLWEBDAV**: A proprietary algorithm identical to SASL/MD5 which takes the user name in the format `username@realm`.
- **ORCLLM**: Oracle's representation of the SMBLM algorithm. The SMBLM algorithm is Oracle's representation of the LM variant of the SMB/CIFS challenge/response authentication algorithm.
- **ORCLNT**: Oracle's representation of the SMBNT algorithm. The SMBNT algorithm is Oracle's representation of the NT variant of the SMB/CIFS challenge/response authentication algorithm.

During Oracle application installation, the Oracle Universal Installer creates for that application a password verifier profile entry containing all the necessary password verification information. It places this entry, as shown in Figure 20-1, immediately below the application entry, which resides under the products entry, which, in turn, resides under the realm-specific Oracle Context.
This verifier profile entry is applicable to users in the specified realm only. For verifier generation to take effect, you must set the `orclcommonusersearchbase` attribute in the common entry of the realm-specific Oracle context to the appropriate value.

*Figure 20–1  Location of the Password Verifier Profile Entry*

**Attributes for Storing Password Verifiers**

Both the directory and Oracle components store the user password in the user entry, but in different attributes. Whereas the directory stores user passwords in the `userPassword` attribute, Oracle components store user password verifiers in the `authPassword`, `orclPasswordVerifier`, or `orclpassword` attribute. Table 20–1 describes each of the attributes used by Oracle components.

**Table 20–1  Attributes for Storing Password Verifiers in User Entries**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>authPassword</code></td>
<td>Attribute for storing a password to an Oracle component when that password is the same as that used to authenticate the user to the directory, namely, <code>userpassword</code>. The value in this attribute is synchronized with that in the <code>userpassword</code> attribute.</td>
</tr>
<tr>
<td></td>
<td>Several different applications can require the user to enter the same clear text password used for the directory, but each application may hash it with a different algorithm. In this case, the same clear text password can become the source of several different password verifiers.</td>
</tr>
<tr>
<td></td>
<td>This attribute is multivalued and can contain all the other verifiers that different applications use for this user's clear text password. If the <code>userpassword</code> attribute is modified, then the <code>authpasswords</code> for all applications are regenerated.</td>
</tr>
<tr>
<td><code>orclPasswordVerifier</code></td>
<td>Attribute for storing a password to an Oracle component when that password is different from that used to authenticate the user to the directory, namely, <code>userpassword</code>. The value in this attribute is not synchronized with that in the <code>userpassword</code> attribute.</td>
</tr>
<tr>
<td></td>
<td>Like <code>authPassword</code>, this attribute is multivalued and can contain all the other verifiers that different applications use for this user's clear text password.</td>
</tr>
</tbody>
</table>
Each of these attribute types has `appID` as an attribute subtype. This attribute subtype uniquely identifies a particular application. For example, the `appID` can be the `ORCLGUID` of the application entry. This attribute subtype is generated during application installation.

In Figure 20–2 on page 20-7, various Oracle components store their password verifiers in Oracle Internet Directory. Oracle Application Server Single Sign-On uses the same password as that for the directory, and hence stores it in the `authPassword` attribute. The other applications use different passwords and hence store their verifiers in `orclPasswordVerifier` attribute.

The following is an example of an application-specific verifier profile entry. Any application that chooses not to use the common verifier framework must create its own verifier profile entry, similar to the one given in the following example. The `orclappid` will be set to the GUID of the application container and it will also be used as a subtype in the verifier attributes `authpassword` and `orclpasswordverifier`.

```
dn: cn=IFSVerifierProfileEntry,cn=IFS,cn=Products,cn=OracleContext,o=Oracle,dc=com
objectclass:top
objectclass:orclpwdverifierprofile
cn:IFSVerifierProfileEntry
orclappid:8FF2DFD8203519C0E034080020C34C50
orclpwdverifierparams;authpassword: crypto:SASL/MDS $ realm:dc=com
orclpwdverifierparams;orclpasswordverifier: crypto:ORCLLM
orclpwdverifierparams;authpassword: crypto:ORCLWEBDAV $ realm:dc=com
$ usernameattribute: mail
$ usernamecase: lower
$ nodomain: TRUE
```

SASL/MD5 and ORCLWEBDAV verifiers are generated by using user name, realm, and password. The user name attribute to be used can be specified in the verifier profile entry. The case of the user name can also be specified as either upper or lower. The ORCLWEBDAV verifier is generated by appending the name of the identity management realm to the user name. If this is not required, then the verifier profile entry must specify `nodomain: TRUE`.

In the previous example, ORCLWEBDAV verifier is generated by using the value of the `mail` attribute without appending the name of the realm. Also, the user name is converted to lower case before generating the verifier.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>orclPassword</code></td>
<td>Attribute for storing only the 03LOGON verifier for enterprise users. The 03LOGON verifier is synchronized with the <code>userpassword</code> attribute, and it is generated by default for all user entries associated with the <code>orcluserv2</code> object class. When Oracle Internet Directory is installed, a database security profile entry is created by default in the Root Oracle Context. The presence of this entry triggers the generation of 03LOGON verifiers for user entries associated with the <code>orcluserv2</code> object class.</td>
</tr>
</tbody>
</table>

Table 20–1 (Cont.) Attributes for Storing Password Verifiers in User Entries
Default Verifiers for Oracle Components

To save you from having to create a profile for each Oracle component, and to enable sharing of password verifiers across all components, Oracle Internet Directory provides a default set of password verifiers. The default verifier types are MD5, MD5-IFS (SASL/MD5 with the user name set to the value of the nickname attribute and realm = Authorized_Users), WEBDAV, ORCLLM, and ORCLNT.

Two profile entries are required: one for applications using personal identification numbers (PINs), which use numeric values only, and another for applications using alphanumeric passwords.

The verifiers for PIN-based applications—for example, the voice mail application in OracleAS Unified Messaging—are stored in the `orclpasswordverifier;orclcommonpin` attribute. The subtype `orclcommonpin` is used to distinguish numeric PINs from alphanumeric passwords. Any application that uses numeric PINs can directly query or compare against the attribute `orclpasswordverifier;orclcommonpin`.

The verifiers for alphanumeric password-based applications—for example, Oracle Internet File System—can be stored in either:

- The `authpassword;orclcommonpwd` attribute—If an application requires its verifier to be synchronized with the `userpassword` attribute
- The `orclpasswordverifier;orclcommonpwd` attribute—If synchronization with the `userpassword` attribute is not required
The subtype `orclcommonpwd` is used to distinguish alphanumeric passwords from numeric PINs. The verifier attributes subtyped by `orclcommonpwd` can be queried against.

These profile entries also contain the list of subscribed applications and these are specified as values in the `uniquemember` attribute in the profile entries. By default, the DN of the Oracle Application Server Single Sign-On identity is one of the subscribed applications. This means that Oracle Application Server Single Sign-On is a proxy member for all its partner applications. All applications not based on Oracle Application Server Single Sign-On must add their identities (DNs) to the `uniquemember` attribute in the appropriate profile entry.

The following is an example of the profile entries.

```
Cn=defaultSharedPwdProfileEntry, cn=common, cn=products, cn=oraclecontext
Objectclass: orclpwdverifierprofile
Cn: orclcommonpwdprofileentry
Orclappid: orclcommonpwd
Orclpwdverifierparams;authpassword: crypto:SASL/MD5 $ realm:Authorized_Users
Orclpwdverifierparams;authpassword: crypto:ORCLWEBDAV $ realm:Authorized_Users
Orclpwdverifierparams;authpassword: crypto:ORCLLM
Orclpwdverifierparams;authpassword: crypto:ORCLOUT
Orclpwdverifierparams;orclpasswordverifier: crypto:SSHA
Uniquemember: cn=SSO,cn=Products,cn=OracleContext
Uniquemember: cn=IFS,cn=Products,cn=OracleContext

Cn=defaultSharedPINProfileEntry, cn=common, cn=products, cn=oraclecontext
Objectclass: orclpwdverifierprofile
Cn: orclcommonpinprofileentry
Orclappid: orclcommonpin
Orclpwdverifierparams;orclpasswordverifier: crypto:MD5
Orclpwdverifierparams;orclpasswordverifier: crypto:SSHA
Uniquemember: cn=SSO,cn=Products,cn=OracleContext
Uniquemember: cn=Unified Messaging,cn=Products,cn=OracleContext
```

For PIN-based applications, `authpassword` is not an option. Such applications use the `orclpasswordverifier` attribute.

**Example: How Password Verification Works for an Oracle Component**

Figure 20–3 shows an example of password verification for an Oracle component. In this example, the Oracle component stores its password verifiers in the directory.
Storing and Managing Password Verifiers for Authenticating to Oracle Components

Figure 20–3 How Password Verification Works

1. The user tries to log in to an application by entering a user name and a clear text password.

2. The application sends the clear text password to the directory server. If the application stores password verifiers in the directory, then the application requests the directory server to compare this password value with the corresponding one in the directory.

3. The directory server:
   a. Generates a password verifier by using the hashing algorithm specified for the particular application
   b. Compares this password verifier with the corresponding password verifiers in the directory. For the compare operation to be successful, the application must provide its appID as the subtype of the verifier attribute. For example:

   ```
   ldapcompare -p389 -D "DN_of_the_application_entity" -w "password" \\
   -b "DN_of_the_user" -a orclpasswordverifier; appID \\
   -v password_of_the_user
   ```
   c. Notifies the application of the results of the compare operation.

4. Depending on the message from the directory server, the application either authenticates the user or not.

If an application does not use the compare operation, then it:

1. Hashes the clear text password entered by the user
2. Retrieves from the directory the hashed value of the clear text password as entered by the user
3. Initiates a challenge to the user to which the client responds. If the response is correct, then the application authenticates the user.

Managing Password Verifier Profiles for Oracle Components by Using Oracle Directory Manager

You can use Oracle Directory Manager to view and modify password verifier profile entries.
Viewing and Modifying a Password Verifier Profile for an Oracle Component by Using Oracle Directory Manager

To view an application’s password verifiers:

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

2. Select Password Verifier Management. The right pane displays two columns:
   - Path to Password Verifier Entry column lists the full DN of each password verifier profile entry
   - Password Verifier Entry column lists the corresponding RDNs of each password verifier profile entry

3. Choose the password verifier you want to view. This displays the Password Verifier Profile dialog box for that password verifier. The fields in this dialog box are described in Table A–14 on page A-8.

4. To modify the hashing algorithm used to generate a password verifier, in the Password Verifier Profile dialog box, enter the new value in the Oracle Password Parameters field.

Managing Password Verifier Profiles for Oracle Components by Using Command-Line Tools

You can view and modify password verifier profiles by using command-line tools.

Viewing a Password Verifier Profile by Using Command-Line Tools

To view an application’s password verifier, perform a search specifying the DN of the password verifier profile.

Example: Modifying a Password Verifier Profile by Using Command-Line Tools

This example changes the hashing algorithm in an application password verifier profile entry. This password verifier synchronizes with the user’s directory password.

```bash
ldapmodify -p 389 -h my_host -v <<EOF
dn: cn=MyAppVerifierProfileEntry,cn=MyApp,cn=Products,cn=OracleContext,o=my_company,dc=com
changetype: modify
replace: orclPwdVerifierParams
orclPwdVerifierParams;authPassword: crypto:SASL/MD5 $ realm:dc=com
EOF
```

Verifier Generation Using Dynamic Parameters

The password verifiers described previously are static password verifiers. That is, they are generated with preconfigured parameters, typically during application installation. Some applications, including Oracle Calendar, Oracle Email, and Oracle Wireless and Voice, require Oracle Internet Directory to generate dynamic password verifiers.

This section contains these topics:

- Generating Dynamic Password Verifiers
- Configuring Oracle Internet Directory to Generate Dynamic Password Verifiers
Generating Dynamic Password Verifiers

Oracle Internet Directory generates a dynamic password verifier when an application requests one. Dynamic verifiers are based on application parameters that are not available until run time.

In order to generate a dynamic password verifier, Oracle Internet Directory needs a user password that was previously stored in a reversible encrypted format. Oracle Internet Directory stores such values in the operational attributes `orclrevpwd` and `orclunsyncrevpwd`. Encrypted values based on `userpassword` are stored in the parameter `orclrevpwd`. Encrypted values based on passwords other than `userpassword`, such as the numeric PINs used by Oracle Calendar, are stored in the parameter `orclunsyncrevpwd`.

Configuring Oracle Internet Directory to Generate Dynamic Password Verifiers

If you are deploying applications that use `userpassword` and that need dynamic password verifiers, you must ensure that Oracle Internet Directory generates the `orclrevpwd` parameter. Oracle Internet Directory generates the attribute `orclrevpwd` when you provision a user if the attribute `orclpwdencryptionenable` in the realm password policy entry is set to 1. Therefore, you must set `orclpwdencryptionenable` to 1 before you provision users. Alternatively, if users were provisioned before you set `orclpwdencryptionenable`, all users must reset their user passwords to trigger the generation of the encrypted value.

If you are deploying applications that use a numeric PIN and that need dynamic password verifiers, you must ensure that Oracle Internet Directory can use the crypto type `3DES` in order to generate the value stored in `orclunsyncrevpwd`. You must specify `3DES` as a value of the attribute `orclpwdverifierparams;orclpasswordverifier` in the common verifier profile entry under the root oracle context. The default DN of this entry is `cn=DefaultSharedPINProfileEntry,cn=Common,cn=Products,cn=Oraclecontext`. To set the value, you would specify:

dn: cn=DefaultSharedPinProfileEntry, cn=Common,
    cn=Products, cn=Oraclecontext
cn: DefaultSharedPinProfileEntry
orclappid: orclcommonpin
orclpwdverifierparams;orclpasswordverifier: crypto:MD5
orclpwdverifierparams;orclpasswordverifier: crypto:3DES
Delegation of Privileges for an Oracle Technology Deployment

This chapter explains how to store all the data for users, groups, and services in one repository, and delegate the administration of that data to various administrators. It also explains the default security configuration in Oracle Internet Directory.

This chapter contains these topics:

- Delegation in the Oracle Identity Management Model
- Overview: Privileges for Administering the Oracle Technology Stack
- Delegation of Privileges for User and Group Management
- Delegation of Privileges for Deployment of Oracle Components
- Delegation of Privileges for Component Runtime

Delegation in the Oracle Identity Management Model

Oracle Identity Management enables you to store all the data for users, groups, and services in one repository, and to delegate a particular administrator for each set of data. By providing both a centralized repository and customized delegated access, Oracle Identity Management is both secure and scalable.

This section contains these topics:

- How Delegation Works
- Delegation in an Oracle Application Server Environment
- About the Default Configuration
- Overview: Privileges for Administering the Oracle Technology Stack

How Delegation Works

Using the delegation model, a global administrator can delegate to realm administrators the privileges to create and manage the identity management realms for hosted companies. Realm administrators can, in turn, delegate to end users and groups the privileges to change their application passwords, personal data, and preferences. Each type of user can thus be given the appropriate level of privileges.

To delegate the necessary privileges, you assign the user to the appropriate administrative group. For example, suppose that you store data for both enterprise users and the e-mail service in the directory, and need to specify a unique administrator for each set of data. To specify a user as the administrator of enterprise
users, you assign that user to, say, the Enterprise User Administrators Group. To specify a user as the administrator of the e-mail services, you assign that user to, say, the E-mail Service Administrators Group.

**Delegation in an Oracle Application Server Environment**

Figure 21–1 shows the flow of delegation in an Oracle Application Server environment.

**Figure 21–1 Delegation Flow in an Oracle Application Server Environment**

As Figure 21–1 on page 21-2 shows, in an Oracle Application Server environment the directory super user (cn=orcadmin) creates:

- The Oracle Context
- The realm
- The realm-specific Oracle Context
- The entry for the realm administrator (cn=orcladmin, cn=users,Enterprise DN)
The realm administrator, in turn, delegates administration of the Oracle Context to specific users by assigning those users to the Oracle Context Administrators Group. Oracle Context Administrators then delegate administration of the Oracle Application Server to one or more users by assigning them to the Oracle Application Server Administrators Group. The Oracle Application Server Administrators install and administer Oracle Application Server components and delegate administration of user and group data to the User and Group Administrators group. The User and Group Administrators create users and groups. They can also grant user and group administrator privileges to other users.

About the Default Configuration

When you first install Oracle Internet Directory, the default configuration establishes access control policies at various points in the directory information tree (DIT). Default access controls are placed on the User and Group containers as described later in this chapter. Likewise, default privileges for specific directory entities are discussed later in this chapter. In addition, certain default privileges are granted to Everyone and to each user as described in Table 21–2.

Table 21–1 Default Privileges Granted to Everyone and to Each User

<table>
<thead>
<tr>
<th>Subject</th>
<th>Default Privileges</th>
</tr>
</thead>
<tbody>
<tr>
<td>Everyone</td>
<td>The following privileges at the Root DSE:</td>
</tr>
<tr>
<td></td>
<td>■ Permission to browse user entries</td>
</tr>
<tr>
<td></td>
<td>■ Search, read, and compare access for all user attributes except the following userpkcs12, orcluserpkcs12hint, userpassword, orclpassword, and orclpasswordverifier</td>
</tr>
<tr>
<td>Each user</td>
<td>Complete access to his or her own attributes—including the userpassword, orclpassword, and orclpasswordverifier attributes.</td>
</tr>
</tbody>
</table>

You can customize this default configuration to meet the security requirements of your enterprise.

Overview: Privileges for Administering the Oracle Technology Stack

Administering the Oracle technology stack requires the privileges described in Table 21–2.
Administrative privileges are delegated to either Oracle components that use the identity management infrastructure or to end users themselves. A privilege can be delegated to either an identity—for example, a user or application—or to a role or group.

This section contains these topics:

- How Privileges Are Granted for Managing User and Group Data
- Default Privileges for Managing User Data
- Default Privileges for Managing Group Data

### How Privileges Are Granted for Managing User and Group Data

To delegate administrative privileges, the Oracle Internet Directory super user does the following:

1. Creates an identity management realm
2. Identifies a special user in that realm who is called the realm administrator

3. Delegates all privileges to that realm administrator

This realm administrator, in turn, delegates certain privileges that Oracle components require to the Oracle defined roles—for example, Oracle Application Server administrators. The Oracle components receive these roles when they are deployed.

In addition to delegating privileges to roles specific to Oracle components, the realm administrator can also define roles specific to the deployment—for example, a role for help desk administrators—and grant privileges to those roles. These delegated administrators can, in turn, grant these roles to end users. In fact, because a majority of user management tasks involve self-service—like changing a phone number or specifying application-specific preferences—these privileges can be delegated to end users by both the realm administrator and Oracle component administrators.

In the case of a group, one or more owners—typically end users—can be identified. If they are granted the necessary administrative privileges, then these owners can manage the group by using Oracle Internet Directory Self-Service Console, Oracle Directory Manager, or command-line tools.

Default Privileges for Managing User Data

Managing users involves privileges to:

- Create and delete user entries
- Modify user attributes
- Delegate user administration to other users

The access control policy point (ACP) for creating users is at the Users container in the identity management realm.

This section describes each of these privileges in more detail.

Creating Users for a Realm

To create users for a realm, an administrator must be a member of the Subscriber DAS Create User Group. Table 21–3 describes the characteristics of this group.

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default ACP</td>
<td>The ACL at the Users container in the default realm allows the Subscriber DAS Create User Group in the realm Oracle Context to create users under the Users container.</td>
</tr>
<tr>
<td>Administrators</td>
<td>The Oracle Internet Directory super user&lt;br&gt;Members of the Oracle Context Administrators Group&lt;br&gt;Members of the User Privilege Assignment Group&lt;br&gt;Members of the DAS Administrators Group&lt;br&gt;Owners of this group</td>
</tr>
<tr>
<td>DN</td>
<td>cn=oracleDASCreateUser,cn=groups,Oracle_Context_DN.</td>
</tr>
</tbody>
</table>

Modifying Attributes of a User

To modify user attributes, an administrator must be a member of the Subscriber DAS Edit User Group. Table 21–4 describes the characteristics of this group.
Delegation of Privileges for User and Group Management

**Table 21–4 Characteristics of the Subscriber DAS Edit User Group**

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default ACP</td>
<td>The ACL at the Users container in the default identity management realm allows the Subscriber DAS Edit User Group in the realm Oracle Context to modify various attributes of users.</td>
</tr>
</tbody>
</table>
| Administrators | The Oracle Internet Directory super user  
                      Members of the Oracle Context Administrators Group  
                      Members of the User Privilege Assignment Group  
                      Members of the DAS Administrators Group  
                      Owners of this group |
| DN            | cn=oracleDASEditUser,cn=groups,Oracle_Context_DN |

**Deleting a User**

To delete a user in a realm, an administrator must be a member of the DAS Delete User Group. Table 21–5 describes the characteristics of this group.

**Table 21–5 Characteristics of the DAS Delete User Group**

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default ACP</td>
<td>The ACL at the Users container in the default identity management realm allows the DAS Delete User Group in the realm Oracle Context to delete a user from the realm.</td>
</tr>
</tbody>
</table>
| Administrators | The Oracle Internet Directory super user  
                      Members of the Oracle Context Administrators Group  
                      Members of the User Privilege Assignment Group  
                      Members of the DAS Administrators Group  
                      Owners of this group |
| DN            | cn=oracleDASDeleteUser,cn=groups,Oracle_Context_DN |

**Delegating User Administration**

A delegated administrator can perform specified operations within the directory and requires permission to add any user to the User Creation, User Edit, or User Delete Groups described previously.

To grant user administration privileges to a delegate administrator, the granting administrator must be a member of the User Privilege Assignment Group. Table 21–6 describes the characteristics of this group.

**Table 21–6 Characteristics of the User Privilege Assignment Group**

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default ACP</td>
<td>The ACL policy for each of the groups previously mentioned allows members of the User Privilege Assignment Group to add users to or remove them from those groups.</td>
</tr>
</tbody>
</table>
| Administrators | The Oracle Internet Directory super user  
                      Oracle Context Administrators Group  
                      Owners of this group. The DNs of these owners are listed as values of the owner attribute in the group. |
| DN            | cn=oracleDASUserPriv,cn=groups,Oracle_Context_DN |
Delegation of Privileges for User and Group Management

Default Privileges for Managing Group Data

Managing users and groups involves privileges to:

- Create and delete group entries
- Modify group attributes
- Delegate group administration to other users

The ACP for creating groups is at the Groups container in the identity management realm.

Creating Groups

To create groups in Oracle Internet Directory, an administrator must be a member of the Group Creation Group. Table 21–7 describes the characteristics of this group.

<table>
<thead>
<tr>
<th>Table 21–7</th>
<th>Characteristics of the Group Creation Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Characteristic</td>
<td>Description</td>
</tr>
<tr>
<td>Default ACP</td>
<td>The ACL at the Groups container in the realm allows the Group Creation Group to add new groups in the realm.</td>
</tr>
<tr>
<td>Administrators</td>
<td>The Oracle Internet Directory super user Members of the Oracle Context Administrators Group Members of the Oracle Application Server Administrators Group Members of the Group Privilege Assignment Group Members of the DAS Administrators Group Owners of this group</td>
</tr>
<tr>
<td>DN</td>
<td>cn=oracleDASCreateGroup,cn=groups,Oracle_Context_DN</td>
</tr>
</tbody>
</table>

Modifying the Attributes of Groups

To modify the attributes of groups under the Groups container in a realm, an administrator must be a member of the Group Edit Group. Table 21–8 describes the characteristics of this group.

<table>
<thead>
<tr>
<th>Table 21–8</th>
<th>Characteristics of the Group Edit Group</th>
</tr>
</thead>
<tbody>
<tr>
<td>Characteristic</td>
<td>Description</td>
</tr>
<tr>
<td>Default ACP</td>
<td>The ACL at the Groups container in the realm allows the Group Edit Group to modify various attributes of groups in the realm.</td>
</tr>
<tr>
<td>Administrators</td>
<td>The Oracle Internet Directory super user Members of the Oracle Context Administrators Group Members of the Oracle Application Server Administrators Group Members of Group Privilege Assignment Group Members of the DAS Administrators Group Owners of this group</td>
</tr>
<tr>
<td>DN</td>
<td>cn=oracleDASEditGroup,cn=groups,Oracle_Context_DN</td>
</tr>
</tbody>
</table>

Deleting Groups

To delete groups, an administrator must have membership in the Group Delete Group. Table 21–9 describes the characteristics of this group.
Delegation of Privileges for Deployment of Oracle Components

### Table 21–9 Characteristics of the Group Delete Group

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default ACP</td>
<td>The ACL at the Groups container in the realm allows the Group Delete Group to delete groups in the realm.</td>
</tr>
</tbody>
</table>
| Administrators| The Oracle Internet Directory super user  
Members of the Oracle Context Administrators Group  
Members of the Group Privilege Assignment Group  
Members of the DAS Administrators Group  
Owners of this group |
| DN            | cn=oracleDASDeleteGroup,cn=groups,Oracle_Context_DN |

**Delegating Group Administration**

To delegate group administration to other users—that is, to add or remove users from the Group Creation, Group Edit, or Group Delete Groups described previously—an administrator must be a member of the Group Privilege Assignment Group. Table 21–10 describes the characteristics of this group.

### Table 21–10 Characteristics of the Group Privilege Assignment Group

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default ACP</td>
<td>The ACL policy for the Group Creation, Group Edit, or Group Delete Groups allows members of Group Privilege Assignment Group to add users to or remove them from those groups.</td>
</tr>
</tbody>
</table>
| Administrators| The Oracle Internet Directory super user  
Members of the Oracle Context Administrators Group  
Owners of the group. The DNs of these owners are listed as values of the owner attribute in the group. |
| DN            | cn=oracleDASUserPriv,cn=groups,Oracle_Context_DN |

**Delegation of Privileges for Deployment of Oracle Components**

This section discusses the groups responsible for deploying Oracle components. It describes the tasks these administrators perform and the privileges they can grant. It includes these topics:

- How Deployment Privileges Are Granted
- Oracle Application Server Administrators
- User Management Application Administrators
- Trusted Application Administrators
Delegation of Privileges for Deployment of Oracle Components

**Note:** Oracle Internet Directory super users have all the privileges of Oracle Application Server Administrators and Trusted Application administrators, and must be members of the Oracle Application Server Administrators Group. They can:

- Assign the Oracle Application Server Administrator role to a user
- Assign the Trusted Application role to a user
- Assign the User Management Application Administrator role to a user

**How Deployment Privileges Are Granted**

To enable administrators to deploy Oracle components, the super user:

1. Grants certain deployment privileges to various groups—for example, the Oracle Application Server Administrators Group
2. Adds the administrators to those privileged groups

The delegated administrators, in turn, can delegate privileges to other administrators.

**Oracle Application Server Administrators**

Table 21–11 describes the characteristics of the Oracle Application Server Administrators Group.

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tasks</td>
<td>Perform repository database installation that creates a repository database registration entry in the directory</td>
</tr>
<tr>
<td></td>
<td>Perform mid-tier installation. To associate a mid-tier with a repository, the user must have the appropriate privileges with a specific repository database.</td>
</tr>
<tr>
<td></td>
<td>Install and configure Oracle Application Server components that create application entities in Oracle Internet Directory</td>
</tr>
<tr>
<td></td>
<td>Grant to component entities the runtime privileges listed later in this section</td>
</tr>
<tr>
<td></td>
<td>Configure provisioning profiles for components so that the components can receive update notifications</td>
</tr>
<tr>
<td>Privileges this group can delegate to components</td>
<td>Read Common User Attributes—except passwords, certificates, and similar security credentials</td>
</tr>
<tr>
<td></td>
<td>Read common group attributes</td>
</tr>
<tr>
<td></td>
<td>Create, edit, and delete groups</td>
</tr>
<tr>
<td></td>
<td>Authenticate a user</td>
</tr>
<tr>
<td></td>
<td>Read application verifiers</td>
</tr>
<tr>
<td>Administrators</td>
<td>Oracle Internet Directory super user</td>
</tr>
<tr>
<td></td>
<td>Oracle Context Administrator</td>
</tr>
<tr>
<td></td>
<td>Owners of this group</td>
</tr>
<tr>
<td>DN</td>
<td>cn=IASAdmins,cn=groups,Oracle_Context_DN</td>
</tr>
</tbody>
</table>
User Management Application Administrators

User Management Application Administrators must be members of the Oracle Application Server Administrators Group.

Table 21–12 describes the characteristics of the User Management Application Administrators Group.

Table 21–12 Characteristics of the User Management Application Administrators Group

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tasks</td>
<td>User Management Application administrators install specific applications that have interfaces to perform user management operations—for example, OracleAS Portal and Oracle Application Server Wireless.</td>
</tr>
<tr>
<td>Privileges this group can delegate to components</td>
<td>Create, edit, and delete user attributes</td>
</tr>
<tr>
<td>Administrators</td>
<td>Oracle Internet Directory super user</td>
</tr>
<tr>
<td></td>
<td>Oracle Context Administrator</td>
</tr>
<tr>
<td></td>
<td>Owners of this group</td>
</tr>
<tr>
<td>DN</td>
<td><code>cn=IAS &amp; User Mgmt Admins,cn=groups, Oracle_Context_DN</code></td>
</tr>
</tbody>
</table>

Trusted Application Administrators

Trusted Application administrators must be members of the Oracle Application Server Administrators Group.

Table 21–13 describes the characteristics of the Trusted Application Administrators Group.

Table 21–13 Characteristics of the Trusted Application Administrators Group

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tasks</td>
<td>Install specific identity management components—for example, Oracle Application Server Single Sign-On, Oracle Delegated Administration Services, and Oracle Application Server Certificate Authority</td>
</tr>
<tr>
<td>Privileges this group can delegate to components</td>
<td>Read, compare, or reset the user password</td>
</tr>
<tr>
<td></td>
<td>Proxy as the end-user</td>
</tr>
<tr>
<td></td>
<td>Read, compare, or modify the user’s certificate and SMIME certificate</td>
</tr>
<tr>
<td>Administrators</td>
<td>Oracle Internet Directory super user</td>
</tr>
<tr>
<td></td>
<td>Oracle Context Administrator</td>
</tr>
<tr>
<td></td>
<td>Owners of this group</td>
</tr>
<tr>
<td>DN</td>
<td><code>cn=Trusted Application Admins,cn=groups, Oracle_Context_DN</code></td>
</tr>
</tbody>
</table>

Delegation of Privileges for Component Runtime

Many Oracle components administer user entries in Oracle Internet Directory and need the corresponding privileges. For example:

- When the Oracle Application Server Single Sign-On server authenticates a user, that server:
Connects to Oracle Internet Directory using its own identity
Verifies that the password entered by the user matches that user’s password stored in the directory

To do this, the Oracle Application Server Single Sign-On server needs permission to compare user passwords. To set up the Oracle Application Server Single Sign-On cookie, it needs permission to read user attributes.

To grant access to a user, OracleAS Portal must retrieve that user’s attributes. To do this, it logs in to Oracle Internet Directory as a proxy user, impersonating the user seeking access. It therefore needs the privileges of a proxy user.

In general, Oracle components can require these privileges:

- Read and modify user passwords
- Compare user passwords
- Proxy on behalf of users accessing applications
- Administer the Oracle Context where all Oracle components store their metadata

Most Oracle components ship with a preconfigured set of privileges. You can change these default privileges to satisfy specific business requirements—for example, by removing privileges to create and delete user entries.

**See Also:** Oracle Application Server Security Guide for further information about the component delegation model

This section describes the security privileges required by Oracle components. It contains these topics:

- Default Privileges for Reading and Modifying User Passwords
- Default Privileges for Comparing User Passwords
- Default Privileges for Comparing Password Verifiers
- Default Privileges for Proxying on Behalf of End Users
- Default Privileges for Managing the Oracle Context
- Default Privileges for Reading Common User Attributes
- Default Privileges for Reading Common Group Attributes
- Default Privileges for Reading the Service Registry
- Default Privileges for Administering the Service Registry

**Default Privileges for Reading and Modifying User Passwords**

Reading and modifying user passwords requires administrative privileges on the security-related attributes in the directory—for example, the `userPassword` attribute. It requires membership in the User Security Administrators Group described in Table 21–14.
Default Privileges for Comparing User Passwords

Comparing user passwords requires permission to compare a user's `userPassword` attribute. This operation is performed by components such as Oracle Unified Messaging that authenticate end users by using their passwords stored in Oracle Internet Directory.

Comparing user passwords requires membership in the Authentication Services Group described in Table 21–15.

Table 21–15 Characteristics of the Authentication Services Group

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default ACP</td>
<td>The ACL policy at the Users container in the default identity management realm allows the Authentication Services Group to perform compare operation on the <code>userPassword</code> attribute of users.</td>
</tr>
<tr>
<td>Administrators</td>
<td>The Oracle Internet Directory super user</td>
</tr>
<tr>
<td></td>
<td>Members of the Oracle Context Administrators Group</td>
</tr>
<tr>
<td></td>
<td>Members of the Application Server Administrators Group</td>
</tr>
<tr>
<td></td>
<td>Owners of this group</td>
</tr>
<tr>
<td>DN</td>
<td><code>cn=authenticationServices,cn=groups,Oracle_Context_DN</code></td>
</tr>
</tbody>
</table>

Default Privileges for Comparing Password Verifiers

To compare password verifiers, a user must have permission to compare the `userPassword` attribute. Comparing password verifiers requires membership in the Verifier Services Group described in Table 21–16.

Table 21–16 Characteristics of the Verifier Services Group

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Administrators</td>
<td>The Oracle Internet Directory super user</td>
</tr>
<tr>
<td></td>
<td>Members of the Oracle Context Administrators group</td>
</tr>
<tr>
<td></td>
<td>Members of the Application Server Administrators group</td>
</tr>
<tr>
<td></td>
<td>Owners of this group</td>
</tr>
<tr>
<td>DN</td>
<td><code>cn=verifierServices,cn=groups,Oracle_Context_DN</code></td>
</tr>
</tbody>
</table>
Default Privileges for Proxying on Behalf of End Users

A **proxy user** has the privilege to impersonate an end user, performing on that user’s behalf those operations for which that user has privileges. In an Oracle Application Server environment, the Oracle Delegated Administration Services proxies on behalf of the end user, and, through the Oracle Internet Directory Self-Service Console, performs operations on that user’s behalf. In such a case, the access controls on the directory server eventually govern the operations that the user can perform.

Proxying on behalf of end users requires membership in the User Proxy Privilege Group described in Table 21–17.

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default ACP</td>
<td>The ACL at the Users container in the default identity management realm allows User Proxy Privilege Group to proxy on behalf of the end user.</td>
</tr>
<tr>
<td>Administrators</td>
<td>The Oracle Internet Directory super user</td>
</tr>
<tr>
<td></td>
<td>Owners of the groups. The DNs of these owners are listed as values of the owner attribute in the group or members of the Oracle Application Server Administrators Group.</td>
</tr>
<tr>
<td></td>
<td>Members of the Trusted Application Administrators Group</td>
</tr>
<tr>
<td>DN</td>
<td><code>cn=userProxyPrivilege,cn=groups,OracleContextDN</code></td>
</tr>
</tbody>
</table>

Default Privileges for Managing the Oracle Context

To manage a specific Oracle Context, a user must have complete access to it. Managing an Oracle Context requires membership in the Oracle Context Administrators Group described in Table 21–18. An Oracle Context Administrators Group exists for each Oracle Context and has administrative permission in the specific Oracle Context.

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default ACP</td>
<td>The ACL policy at the root node of the Oracle Context allows members of Oracle Context Administrators Group to perform all administrative operations within the Oracle Context. Such a policy is set up when a new Oracle Context is created in the directory.</td>
</tr>
<tr>
<td>Administrators</td>
<td>The Oracle Internet Directory super user</td>
</tr>
<tr>
<td></td>
<td>Members of the Oracle Context Administrators Group</td>
</tr>
<tr>
<td>DN</td>
<td><code>cn=oracleContextAdmins,cn=groups,Oracle_Context_DN</code></td>
</tr>
</tbody>
</table>

Default Privileges for Reading Common User Attributes

Common user attributes are: `mail`, `orclguid`, `displayname`, `preferredlanguage`, `orcltime`, `gender`, `dateofbirth`, `telephonenumber`, `wirelessaccountnumber`. To read these attributes requires membership in the Common User Attributes Group described in Table 21–19.
Default Privileges for Reading Common Group Attributes

Common group attributes are: cn, uniquemember, displayname, and description. To read these attributes requires membership in the Common Group Attributes Group described in Table 21–20 on page 21-14.

Table 21–20 Characteristics of the Common Group Attributes Group

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default ACP</td>
<td>The default ACL is on the Group container in the realm and grants permission to read these attributes: cn, uniquemember, displayname, and description.</td>
</tr>
<tr>
<td>Administrators</td>
<td>The Oracle Internet Directory super user Members of the Application Server Administrators Group Owners of this group</td>
</tr>
<tr>
<td>DN</td>
<td>cn=commongroupattributes,cn=groups,Oracle_Context_DN</td>
</tr>
</tbody>
</table>

Default Privileges for Reading the Service Registry

To view the contents of the Service Registry requires membership in the Service Registry Viewers Group described in Table 21–21 on page 21-14.

Table 21–21 Characteristics of the Service Registry Viewers Group

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default ACP</td>
<td>The default ACL is on the Services container in the root Oracle Context.</td>
</tr>
<tr>
<td>Administrators</td>
<td>The Oracle Internet Directory super user Members of the Application Server Administrators Group Owners of this group</td>
</tr>
<tr>
<td>DN</td>
<td>cn=service_registry viewers,cn=services,cn=rootoraclecontext,</td>
</tr>
</tbody>
</table>

Default Privileges for Administering the Service Registry

To administer the Service Registry requires membership in the Service Registry Administrators Group described in Table 21–22 on page 21-14.

Table 21–22 Characteristics of the Common Group Attributes Group

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default ACP</td>
<td>The default ACL is on the Services container in the root Oracle Context.</td>
</tr>
</tbody>
</table>
Table 21–22 (Cont.) Characteristics of the Common Group Attributes Group

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Description</th>
</tr>
</thead>
</table>
| Administrators | The Oracle Internet Directory super user  
                              Members of the Application Server Administrators Group  
                              Owners of this group |
| DN             | cn=service registry  
                              admins,cn=services,cn=rootoraclecontext, |
Part IV
Directory Deployment

This part discusses important deployment considerations. It includes these chapters:

- Chapter 22, "Directory Deployment Considerations"
- Chapter 23, "Deployment of Oracle Identity Management Realms"
- Chapter 24, "Capacity Planning for the Directory"
- Chapter 25, "Tuning Considerations for the Directory"
- Chapter 26, "Garbage Collection in Oracle Internet Directory"
- Chapter 27, "Migration of Data from Other Data Repositories"
- Chapter 28, "Server Chaining"
Directory Deployment Considerations

This chapter discusses issues to consider when deploying Oracle Internet Directory. It helps you assess enterprise directory requirements and make effective deployment choices. Although the recommendations in this chapter are primarily for directories in medium to large enterprises and Internet Service Providers (ISPs), the principles apply to other environments as well.

This chapter contains these topics:

- The Expanding Role of Directories
- Logical Organization Of Directory Information
- Physical Distribution: Partitions, Replicas, and High Availability
- Oracle Directory Integration Platform
- Capacity Planning, Sizing, and Tuning

See Also:

- Chapter 24, "Capacity Planning for the Directory" for more detailed information about capacity planning
- "High Availability and Failover Considerations" in Oracle Application Server High Availability Guide for more detailed information about high availability
- Chapter 25, "Tuning Considerations for the Directory" for more detailed information about tuning
- Oracle Application Server High Availability Guide for information about failover in clustered environments

The Expanding Role of Directories

Today, most enterprises are at various stages of deploying centralized and consolidated LDAP-compliant directories. Some have had non-LDAP-compliant directories—for example, NDS or ISO X.500—and are now converting to the corresponding LDAP-enabled versions. They do this either to accommodate LDAP-reliant Internet clients, such as those embedded in Web browsers, or to consolidate the increasing number of platforms and services that use directories.

The increased numbers of LDAP-enabled applications make availability and performance requirements for LDAP-compliant directories critical. Most environments need to update their deployments.

Enterprises should plan a robust and flexible deployment to accommodate:
The increased volume of information in the directory
The number of applications that rely on the directory
Such load characteristics as concurrent access and throughput
As the directory becomes more central to the operation of the network and its services, deployment choices become critical.

Logical Organization Of Directory Information
Oracle Internet Directory serves as a shared repository for the entire Oracle Identity Management infrastructure. A carefully planned logical structure of the directory enables:

- Enforcement of security policies that meet the requirements of your deployment
- A more efficient physical deployment of the directory service
- Easier configuration of synchronization of a third-party directory with Oracle Internet Directory

See Also: "Planning the Directory Information Tree for Identity Management" on page 23-1

Physical Distribution: Partitions, Replicas, and High Availability
You can distribute directory data in two ways:

- By maintaining the entire directory on one server
- By hosting different naming contexts on different servers and connecting them by using knowledge references

See Also: "Distributed Directories" on page 3-16

This section contains these topics:

- An Ideal Deployment
- Partitioning Considerations
- Replication Considerations
- High Availability Considerations

An Ideal Deployment
Although it would be simpler and more secure to store all naming contexts in one central directory, this central directory would then be a single point of failure.

One solution might be to implement redundant LDAP servers and their associated databases. However, even redundancy might not provide the needed connectivity, accessibility, and performance that most global organizations need at all their regions and sites. These requirements might, in fact, call for replicas physically located at various regions across the corporate geography.

If Oracle Internet Directory supported only single-master configuration, then logical consolidation of the directory would be difficult. Each region or group would want to store the master replica for the naming context on which that group relies. Because administrators would need to use a different data management procedure for each
partition, this could mean a lack of uniformity in the administrative policies among the partitions.

Fortunately, because multimaster replication allows "update anywhere" configurations, it is more efficient and less costly to consolidate the directory rather than to maintain multiple partitions.

Here is a simple and practical recommendation for a robust centralized corporate directory:

■ Establish a network of two or more directory nodes, each holding all the naming contexts. Set up these nodes in a multimaster configuration.
■ Deploy these individual nodes, one in each geographic region, to suit the corporate data network connectivity. For example, if a region is connected to the rest of the network by way of a slow link, then it is better to locate a dedicated directory server for use by the clients in that region.
■ Individually configure each regional server for failover and recovery.

Remember: Even if all the naming contexts are consolidated, you can still achieve administrative autonomy for various logical naming contexts. You do this by establishing appropriate access control policies at the root of each naming context.

See Also: "High Availability Considerations" on page 22-4 for a discussion of redundancy

Partitioning Considerations

A directory with too many partitions generally has more administrative overhead than benefits. This is because each partition requires you to plan backup, recovery, and other data management functions.

Typically, the reasons for maintaining partitions are:

■ They correspond to administrative and data ownership boundaries that are better left independent
■ The enterprise network has regions that are connected with expensive or low-speed links and many partitions have only local access needs
■ The lack of availability of a partition does not have a larger impact
■ Maintaining an entire corporate directory in a certain region is too expensive

When you use partitioning, connect one partition to another by using a knowledge reference.

Note: LDAP does not support automatic chaining of knowledge references by the LDAP server. The majority of client side LDAP APIs support client-driven knowledge reference chasing. However, there is no guarantee that knowledge references will be supported in all the LDAP tools. The lack of consistent knowledge reference support across all available tools is a factor to consider before deciding to use partitions.

Replication Considerations

LDAP directory replication architecture is based on a loose consistency model: Two replicated nodes in a replication agreement are not guaranteed to be consistent in real time. This increases the overall flexibility and availability of the directory network,
because a client can modify data without all interconnected nodes being available. Suppose, for example, that one node is unavailable or heavily loaded. With multimaster replication, the operation can be performed on an alternate node, and all interconnected nodes synchronize in due course.

There are many reasons to implement a replicated network, including the following:

- Local accessibility and performance requirements

  Most corporations have operations in many regions in the world, and those operations need a common directory. Suppose that the regions were interconnected with low bandwidth links involving multiple intermediate routers. A client accessing a directory server from outside the region could experience a very high latency, and even inadequate throughput.

  In such cases, a regional replica—enabled by multimaster replication to receive updates—is essential. Moreover, the replication data transfer can be scheduled for off-peak hours in the underlying Oracle Database Advanced Replication.

- Load balancing

  When directory access exceeds the capacity of an existing server, an additional server must share the load. With Oracle Internet Directory, two such systems can be deployed in a multimaster replication mode. In fact, even when planning the directory deployment to meet a specific estimated load, it can be less costly to maintain two relatively low-end systems than one high-end system. In addition to load balancing, such configurations also contribute to higher system availability.

- Failure tolerance and higher overall system availability

  One of the most important reasons to implement directory replication is to increase overall system availability. When one server is unavailable, the traffic can be routed to other available servers. This can be transparent to clients.

See Also: The section on planning the physical deployment of Oracle Internet Directory in Oracle Identity Management Concepts and Deployment Planning Guide for more information about replicated directory configurations.

High Availability Considerations

Because a directory service has a critical function in an enterprise, deployment should take failure recovery and high availability into consideration. This includes developing backup and recovery strategies for individual nodes.

In addition to multimaster replication, consider the following failover and high-availability options for potential deployment at any Oracle Internet Directory installation:

- Intelligent Client Failover

  All LDAP clients connecting to Oracle Internet Directory can maintain a list of alternate server instances of Oracle Internet Directory to contact if their connection with a given server instance is abruptly broken.

- Intelligent Network Level Failover

  There are several hardware and software solutions that can detect the failure of the system hosting Oracle Internet Directory. These solutions can intelligently reroute future connection requests to an alternate server. Some of these solutions balance the load of incoming connection requests with alternate servers, while also providing the necessary failover capabilities.
Multiple installations of Oracle Internet Directory on one host

You can run more than one installation of Oracle Internet Directory on a single host and then replicate between them. This can be useful in providing up-to-date directory data on the same machine by automatically backing up that data. It also enables you to provide for failover by using only two nodes: If one node fails, then both instances of Oracle Internet Directory can run on the other node.

Because Oracle Internet Directory is a client of the Oracle Database, other failover technologies, such as Oracle Real Application Clusters, are also available.

See Also:
- "High Availability and Failover Considerations" in Oracle Application Server High Availability Guide for further details about high-availability and failover options available with Oracle Internet Directory
- The section on planning the physical deployment of Oracle Internet Directory in Oracle Identity Management Concepts and Deployment Planning Guide for more information about high availability for the directory

Oracle Directory Integration Platform

You can reduce administrative time and costs by integrating your applications and directories—including third-party LDAP directories—with Oracle Internet Directory. Oracle Directory Integration Platform, a component of Oracle Identity Management, enables you to do this. For example, you might need to do the following:

- Keep employee records in Oracle Human Resources consistent with those in Oracle Internet Directory. Oracle Directory Integration Platform 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. Oracle Directory Integration Platform provides this notification through the Oracle Directory Provisioning Integration Service.

Throughout the integration process, the Oracle Directory Integration Platform 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 the Oracle Directory Integration Platform can synchronize the data in Microsoft Active Directory with that in Oracle Internet Directory.

See Also: Oracle Identity Management Integration Guide

Capacity Planning, Sizing, and Tuning

When estimating enterprise-wide and regional requirements for directory usage, plan for future needs. Depending on other configuration choices for replication and failover, there could be more than one directory node, each with its own load and capacity requirements. In this case, you must individually size each directory node.
As an enterprise increases its directory usage, more applications rely on Oracle Internet Directory to serve their requests in a timely manner. Ensure that the Oracle Internet Directory installation can live up to the performance and capacity expectations of those applications.

You can influence the capacity and performance of a given Oracle Internet Directory installation in two phases of the deployment process:

- **Planning phase**
  During this phase, gather the requirements of all directory users and establish a unified performance and capacity requirement. This consists of capacity planning and system sizing.

- **Implementation phase**
  Once you have the hardware, tune the Oracle Internet Directory software stack for best use of the hardware resources. This improves the performance of Oracle Internet Directory and of the LDAP client applications.

This section contains these topics:

- Capacity Planning
- Sizing Considerations
- Tuning Considerations

### Capacity Planning

Capacity planning is the process of determining performance and capacity requirements. You base these on typical models of directory usage in the enterprise.

When trying to estimate the required capacity of an Oracle Internet Directory installation, consider:

- The type of LDAP client applications
- The number of users accessing those applications
- The nature of LDAP operations those applications perform
- The number of entries in the DIT
- The type of operations performed against the Oracle directory server
- The number of concurrent connections to the Oracle directory server
- The peak rate at which operations need to be performed by the Oracle directory server
- The average latency of operations required under peak load conditions

While estimating these details, allow room for future increases in directory usage.

### Sizing Considerations

Once you have established the fundamental capacity and performance requirements, translate them into system requirements. This is called system sizing. Some of the details to consider in this phase are:

- The type and number of CPUs for the Oracle Internet Directory server computer
- The type and size of disk subsystems for the Oracle Internet Directory server computer
The amount of memory required for the Oracle Internet Directory server computer

- The type of network used for LDAP messages from the clients

Based on current experience, Table 22–1 indicates the approximate level of CPU power required for various deployment scenarios for Oracle Internet Directory.

### Table 22–1 CPU Power for Various Deployment Scenarios

<table>
<thead>
<tr>
<th>Usage</th>
<th>Active Connections</th>
<th>Num CPUs</th>
<th>SPECint_rate95 baseline</th>
<th>System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Departmental</td>
<td>0-500</td>
<td>2</td>
<td>60 to 200</td>
<td>Compaq AlphaServer 8400 5/300 (300Mhz x 2)</td>
</tr>
<tr>
<td>Organization wide</td>
<td>500-2000</td>
<td>4</td>
<td>200 to 350</td>
<td>IBM RS/6000 J50 (200MHz x 4)</td>
</tr>
<tr>
<td>Enterprise wide</td>
<td>2000+</td>
<td>4+</td>
<td>350+</td>
<td>Sun Ultra 450 (296 MHz x 4)</td>
</tr>
</tbody>
</table>

The amount of disk space required for an installation of Oracle Internet Directory is directly proportional to the number of entries stored in the DIT. Table 22–2 gives the approximate disk space requirements for variously sized DITs.

### Table 22–2 Approximate Disk Space Requirements for Variously Sized DITs

<table>
<thead>
<tr>
<th>Number of Entries in DIT</th>
<th>Disk Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>100,000</td>
<td>450MB to 650MB</td>
</tr>
<tr>
<td>200,000</td>
<td>850MB to 1.5GB</td>
</tr>
<tr>
<td>500,000</td>
<td>2.5GB to 3.5GB</td>
</tr>
<tr>
<td>1,000,000</td>
<td>4.5GB to 6.5GB</td>
</tr>
<tr>
<td>1,500,000</td>
<td>6.5GB to 10GB</td>
</tr>
<tr>
<td>2,000,000</td>
<td>9GB to 13GB</td>
</tr>
</tbody>
</table>

The data in this table makes the following assumptions:

- There are approximately 20 cataloged attributes
- There are approximately 25 attributes for each entry
- The average size of an attribute is approximately 30 bytes

The amount of memory required for Oracle Internet Directory is mostly governed by the amount of database buffer cache that a deployment site desires. Often, the size of the database buffer cache is directly proportional to the number of entries in the DIT. Table 22–3 on page 22-7 provides estimates of the memory requirements for various DIT sizes.

### Table 22–3 Estimates of the Memory Requirements for Various DIT Sizes

<table>
<thead>
<tr>
<th>Directory Type</th>
<th>Number of Entries</th>
<th>Minimum Memory</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>Less than 600,000</td>
<td>512MB</td>
</tr>
<tr>
<td>Medium</td>
<td>600,000 to 2,000,000</td>
<td>1GB</td>
</tr>
<tr>
<td>Large</td>
<td>Greater than 2,000,000</td>
<td>2GB</td>
</tr>
</tbody>
</table>
Tuning Considerations

Oracle Corporation recommends that you properly tune Oracle Internet Directory before using it in a production environment. Before tuning, ensure that there are adequate testing mechanisms and sample data in the directory to simulate a real world usage scenario. Perhaps you can use the applications that rely on the directory for testing purposes.

Any tool for testing the performance of Oracle Internet Directory must be able to show:

- The overall throughput it is noticing
- The average latency of operations

In this way, the tool provides a feedback mechanism for determining the effects of tuning and providing direction to the overall tuning effort.

Some of the commonly tuned properties of an Oracle Internet Directory installation include:

- CPU usage
  This is determined, to a large extent, by:
  - The number of Oracle directory servers
  - The number of database connections opened by each server
  On the one hand, too large a number of Oracle directory servers and database connections can cause too much contention for available CPU resources. On the other hand, too small a number of Oracle directory servers and database connections can leave much of the CPU power under-utilized. Consider adjusting these numbers to the appropriate levels based on available CPU resources and the expected peak load.

- Memory usage
  The main consumer of memory in an Oracle Internet Directory installation is the database buffer cache, which is part of the SGA. In some cases, allocating a very large database buffer cache can eliminate much disk I/O for Oracle data files. However, it can also cause paging, which is detrimental to performance. Alternatively, having a small database buffer cache causes too much disk I/O, and that is also detrimental to performance. Tune the memory usage of the system so that all consumers of memory in the system can get physical memory without needing to use paging.

- Disk usage
  Because all of the data served by Oracle Internet Directory resides in database tablespaces, pay attention to any tuning that can increase the I/O throughput. Common techniques for disk tuning include:
  - Balancing tablespaces on different logical and physical drives
  - Striping logical volumes onto multiple physical volumes
  - Distributing disk volumes across multiple I/O controllers

See Also: Chapter 24, "Capacity Planning for the Directory" for further details on various tuning tips and techniques
This chapter discusses identity management realms and how to plan and configure them for both enterprise and hosted deployments.

This chapter contains these topics:

- Planning the Directory Information Tree for Identity Management
- Identity Management Realms in an Enterprise Deployment
- Identity Management Realms in a Hosted Deployment
- Identity Management Realm Implementation in Oracle Internet Directory
- Default Directory Information Tree and the Identity Management Realm
- Administration of Identity Management Realms

Planning the Directory Information Tree for Identity Management

Oracle Internet Directory serves as a shared repository for the entire Oracle Identity Management infrastructure. A carefully planned logical structure of the directory enables:

- Enforcement of security policies that meet the requirements of your deployment
- An efficient physical deployment of the directory service
- Easier configuration of synchronization of a third-party directory with Oracle Internet Directory

Figure 23–1 shows a directory information tree for a hypothetical company, called MyCompany, that is deploying identity management.
MyCompany makes the following decisions with respect to the logical organization of the directory in their U.S. deployment:

- A domain name-based scheme is to represent the overall DIT hierarchy. Because the identity management infrastructure is being rolled out in the us domain, the root of the DIT is `dc=us,dc=mycompany,dc=com`.

- Within the naming context chosen, all users are represented under a container called `cn=users`. Within this container, all users are represented at the same level—that is, there is no organization-based hierarchy. In addition, the `uid` attribute is chosen as the unique identifier for all users.

- Within the naming context chosen, all enterprise groups are represented under a container called `cn=groups`. Within this container, all enterprise groups are represented at the same level. The naming attribute for all group entries is `cn`.

- Finally, the container `dc=us` is chosen as the root of the identity management realm. In this case, the name of the realm is `us`. The deployment expects to enforce similar security policies for all users who fall under the scope of the `us` realm.

Planning the logical organization of the directory for Oracle Identity Management involves planning:

- The overall structure of the directory information tree
- The directory containment and naming for users and groups
- The identity management realm

This section discusses further details to consider when designing the logical organization of directory information. It contains these topics:

- Planning the Overall Directory Structure
- Planning the Names and Containment of Users and Groups
- Planning the Identity Management Realm
- Migrating a DIT from a Third-Party Directory
Planning the Overall Directory Structure

This task involves designing the basic directory information tree that all identity management-integrated applications in the enterprise are to use. As you do this, keep these considerations in mind:

- The directory organization should facilitate clean and effective access control. If either full or partial replication is planned, then proper boundaries and policies for replication can be enforced only if the design of the DIT brings out the separation.

- If the enterprise is integrating with a third-party directory server, then it is best to align the DIT design of Oracle Internet Directory with the existing DIT. This consideration also applies to deployments that are rolling out Oracle Internet Directory now but plan to roll out another directory later—for example, Microsoft Active Directory that is required for the operation of software from Microsoft. In each case, choosing an Oracle Internet Directory DIT design that is more consistent with that of the third-party directory makes management of user and group objects easier through Oracle Delegated Administration Services and other middle-tier applications.

- In a single enterprise scenario, choosing a DIT design that aligns with the DNS domain name of the enterprise suffices. For example, if Oracle Internet Directory is set up in a company having the domain name mycompany.com, then a directory structure that has dc=mycompany, dc=com is recommended. Oracle Corporation recommends that you not use departmental or organization level domain components such as engineering in engineering.mycompany.com.

- If the enterprise has an X.500 directory service, and no other third-party LDAP directories in production, then it may benefit by choosing a country-based DIT design. For example, a DIT design with the root of o=mycompany, c=US might be more suitable for enterprises which already have an X.500 directory service.

- Because the directory can be used by several applications—both from Oracle and from third-parties alike—the naming attributes used in relative distinguished names (RDNs) constituting the overall DIT structure should be restricted to well-known attributes. The following attributes are generally well-known among most directory-enabled applications:
  - c: The name of a country
  - dc: A component of a DNS domain name
  - l: The name of a locality, such as a city, county or other geographic region
  - o: The name of an organization
  - ou: The name of an organizational unit
  - st: The name of a state or province

- A common mistake is to design the DIT to reflect either the corporate divisional or organizational structure. Because most corporations undergo frequent reorganization and divisional restructuring, this is not advisable. It is important to insulate the corporate directory from organizational changes as much as possible.

Planning the Names and Containment of Users and Groups

Most of the design considerations that are applicable to the overall DIT design are also applicable to the naming and containment of users and groups. This section offers some additional things to consider when modeling users and groups in Oracle Internet Directory.
Considerations for Users
The Oracle Identity Management infrastructure uses Oracle Internet Directory as the repository for all user identities. Even though a user might have account access to multiple applications in the enterprise, there is only one entry in Oracle Internet Directory representing that user's identity. The location and content of these entries in the overall DIT must be planned before deploying Oracle Internet Directory and other components of the Oracle Identity Management infrastructure.

- As mentioned in the previous section, it is tempting to organize users according to their current departmental affiliations and hierarchy. However, this is not advisable because most corporations undergo frequent reorganization and divisional restructuring. It is more manageable to capture a person's organizational information as an attribute of that person's directory entry.

- There are no performance benefits derived from organizing users in a hierarchy according to organizational affiliations or management chain. Oracle Corporation recommends that you keep the DIT containing users as flat as possible.

- If the deployment has different user populations with each one maintained and managed by a different organization, then Oracle Corporation recommends subdividing users into containers based on these administrative boundaries. This simplifies the setting of access controls and helps in cases where replication is needed.

- The out-of-the-box default nickname attribute for uniquely identifying users in lookup operations is `uid`. This is the default attribute used for logins. The out-of-the-box default naming attribute for constructing a DN is `cn`.

- The default attribute for uniquely identifying users is `cn` or CommonName. The typical value of CommonName is the person's full name. People's names or email addresses, however, can change and therefore might not be suitable as values of this attribute. If possible choose a value that will not change and that uniquely identifies a user, such as the employee id.

- Typically, most enterprises have a Human Resources department that establishes rules for assigning unique names and numbers for employees. When choosing a unique naming component for directory entries, it is good to exploit this administrative infrastructure and use its policies.

- It is required that all user entries created in the directory belong to the following object classes: `inetOrgPerson`, `orclUserV2`.

- If you already have a third-party directory, or plan to integrate with one in the future, then it is beneficial to align the user naming and directory containment in Oracle Internet Directory with the one used in the third-party directory. This simplifies the synchronization and subsequent administration of the distributed directories.

**Note:** In Oracle Internet Directory Release 9.0.2, the default value for the nickname attribute was `cn`. As of Release 9.0.4, the default value for this attribute is `uid`.

Considerations for Groups
Some applications integrated with the Oracle Identity Management infrastructure can also base their authorizations on enterprise-wide groups created by the deployment in Oracle Internet Directory. Like user entries, the location and content of these group
entries should also be carefully planned. When you design groups, consider the following:

- There are no performance benefits to be gained from organizing enterprise groups in a hierarchy based on the organizational affiliations or ownership. Oracle Corporation recommends keeping the DIT that contains groups as flat as possible. This facilitates easy discovery of groups by all applications and fosters sharing of these groups across applications.

- It is preferable to separate the users and groups in the DIT so that different management policies can be applied to each set of entries.

- The attribute used to uniquely identify a group should be `cn` or `CommonName`.

- All group entries created by the enterprise in the directory should belong to the following object classes: `groupOfUniqueNames` and `orclGroup`. The former object class is an internet standard for representing groups. The latter is useful when using the Oracle Internet Directory Self-Service Console to manage groups.

- Instead of creating new directory access controls for each enterprise-wide group, consider doing the following:
  1. Use the `owner` attribute of the group to list which users own this group.
  2. Create an access control policy at a higher level that grants all users listed in the `owner` attribute special privileges to perform the various operations.

- In the `description` attribute, provide information for users to understand the purpose of the group.

- Consider using the `displayName` attribute from the `orclGroup` object class. This enables Oracle Delegated Administration Services and Oracle Internet Directory Self-Service Console to display a more readable name for the group.

- If you have different sets of groups, each of which is maintained and managed by a different organization with its own administrative policies, then sub-divide the groups into containers based on these administrative boundaries. This simplifies the setting of access controls. It also helps when replication is needed.

- If you already have a third-party directory, or plan to integrate with one in the future, then align the group naming and directory containment in Oracle Internet Directory with the one used in the third-party directory. This simplifies the synchronization and subsequent administration of the distributed directories.

### Planning the Identity Management Realm

The previous sections describe guidelines for you to structure the overall DIT and the placement of users and groups for your deployment. Because implementing these guidelines can lead to an infinite number of deployment configurations, you need to capture the intent of your deployment in metadata in the directory itself. This metadata enables Oracle software and other third-party software relying on the Oracle Identity Management infrastructure to understand the deployment intent and successfully function in customized environments.

In Oracle Internet Directory, this deployment intent is captured in the identity management realm. The realm also helps set identity management policies for users and groups whose placement is described in the previous section.

The identity management realm is a well-scoped area in the directory that consists of:

- A well-scoped collection of enterprise identities—for example, all employees in the US
Planning the Directory Information Tree for Identity Management

- A collection of identity management policies associated with these identities
- A collection of groups—that is, aggregations of identities—that makes it easier to set identity management policies

Once you have decided on the overall DIT structure and the placement of users and groups, you need to identify the directory entry to serve as the root of the identity management realm. This entry determines the scope of the identity management policies defined in the realm. By default, the scope is the entire directory subtree under the root of the identity management realm. Under this entry, a special entry called OracleContext is created. It contains the following:

- The deployment-specific DIT design, including user and group naming and placement, as described in previous sections
- The identity management policies associated with this realm
- Additional realm-specific information specific to Oracle applications

When planning the identity management realm, consider the following:

- The security needs of your enterprise must dictate the choice of the root of the identity management realm. Typically, most enterprises need only one realm. However, multiple realms may be required when multiple user populations are managed with different identity management policies.
- If you already have a third-party directory, or plan to integrate with one in the future, then align the choice of the identity management realm root with the DIT design of the third-party directory. This simplifies the synchronization and subsequent administration of the distributed directories.
- To configure and administer identity management realms, use the administrative tools provided by Oracle Internet Directory. These include the Oracle Internet Directory Configuration Assistant, the Oracle Internet Directory Self-Service Console, and command-line tools.
- Once you have used the Oracle Internet Directory tools to configure the identity management realm, plan on updating the directory naming and containment policies to reflect the customizations made by the deployment. This update must happen prior to installing and using other Oracle components that use the Oracle Identity Management infrastructure.

Figure 23–2 shows an example of an identity management realm for an enterprise called MyCompany.
In the example in Figure 23–2, the deployment uses a domain name-based DIT structure. The container `dc=us, dc=mycompany, dc=com` is the root of the identity management realm. This results in the creation of a new identity management realm whose scope, by default, is restricted to the entire directory subtree under the entry `dc=us`. The name of the identity management realm is `US`.

**Migrating a DIT from a Third-Party Directory**

To migrate a DIT from a third-party directory, use the techniques described in *Oracle Identity Management Integration Guide* for synchronizing with third-party metadirectory solutions and integrating with third-party directories. If you are migrating a DIT from a Microsoft Active Directory environment, also see the chapter on integration with the Microsoft Active Directory Environment. Oracle recommends that you configure the Oracle Internet Directory DIT to be identical to the third-party DIT.

**Identity Management Realms in an Enterprise Deployment**

This section discusses deployments with single identity management realms and those with multiple ones. It contains these topics:

- **Single Identity Management Realm in the Enterprise**
- **Multiple Identity Management Realms in the Enterprise**

**Single Identity Management Realm in the Enterprise**

In this scenario, an enterprise has a single set of users, all of whom are managed with the same identity management policies. This is the default configuration of all Oracle products. It includes only one default identity management realm in Oracle Internet Directory and all Oracle components in the enterprise serve users in that realm. *Figure 23–3* illustrates this usage.
Identity Management Realms in a Hosted Deployment

**Figure 23–3  Enterprise Use Case: Single Identity Management Realm**

In the example in Figure 23–3, there is a single, default identity management realm containing employees only. In that realm all users and groups are managed and all share access to the same applications, Application A and Application B.

**Multiple Identity Management Realms in the Enterprise**

You can use the same identity management infrastructure to serve both internal as well as external self-registered users. Because the identity management policies for internal and external users are different, you can deploy two realms, one for internal and one for external users. Figure 23–4 on page 23-8 illustrates this usage.

**Figure 23–4  Enterprise Use Case: Multiple Identity Management Realms**

In the example in Figure 23–4, the default identity management realm is for internal users—namely, employees—and these have access to Applications A, B, and C. The external identity management realm is for external users, and they have access to Applications C and D.

**Identity Management Realms in a Hosted Deployment**

In a hosted deployment, the application service provider (ASP) supplies one or more companies with identity management services and hosts applications for them. Each
hosted company is associated with a separate identity management realm where users of that company are managed. Users belonging to the application service provider are managed in a different realm, typically the default realm.

Figure 23–5 shows a hosted deployment with two hosted companies.

In the example in Figure 23–5, the ASP users are in the default identity management realm. The ASP manages its users, groups and associated policies in that realm. ASP users manage Applications A, B, and C for the hosted companies. Hosted company MyCompany users are in the MyCompany identity management realm. They use Applications A and B. Hosted company XY Corp users are in the XY Corp identity management realm. They use Applications B and C.

Table 23–1 describes the objects in an identity management realm.

<table>
<thead>
<tr>
<th>Object</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Root Oracle Context</td>
<td>This object contains:</td>
</tr>
<tr>
<td></td>
<td>■ A pointer to the default identity management realm in the infrastructure</td>
</tr>
<tr>
<td></td>
<td>■ Information on how to locate a realm given a simple name of the realm</td>
</tr>
<tr>
<td>Identity Management Realm</td>
<td>A normal directory entry with a special object class associated with it.</td>
</tr>
</tbody>
</table>
To make configuration easier, Oracle Internet Directory, at installation, creates a default DIT and sets up a default identity management realm.

As Figure 23–6 shows, the default identity management realm is part of a global DIT. The node just below 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 purposes of illustration, the cn=Users node contains two leaves: uid=user1 and uid=user2. Similarly, the cn=Groups node contains cn=group1 and cn=group2.

Oracle Internet Directory gives you the option of setting up a DIT based on the domain of the computer on which the installation is performed. For example, if the installation is on a computer named oidhost.us.mycompany.com, then the root of the default identity management realm is dc=us, dc=mycompany, dc=com.

It also gives you the option of specifying a different DN that meets your deployment needs as the root of your default identity management realm on the Specify Namespace in Internet Directory install screen. For example, if you plan to integrate...
your Identity Management installation with a third-party directory, it is recommended
that you specify a DN that matches the DN of the default naming context in the
third-party directory. For more details on obtaining the default naming context from a
third-party directory, refer to the chapter on integrating with third-party directories in
the Oracle Identity Management Integration Guide.

During configuration, Oracle Internet Directory creates the following:

- An Oracle Context associated with the default identity management realm. The
Oracle Context stores all the realm-specific policies and metadata. Using the
example in the previous paragraph, it creates the Oracle Context with the
distinguished name `cn=OracleContext,dc=us,dc=mycompany,dc=com`.
This entry and the nodes under it enable Oracle software to detect realm-specific
policies and settings.

- A directory structure and naming policies in the default identity management
realm. These enable Oracle components to locate various identities. The default
values for these are:
  - All users are located in the container `cn=users` under the base of the identity
    management realm. In this example, it is
    `cn=users,dc=us,dc=mycompany,dc=com`.
  - Any new users created in the identity management realm using the Oracle
    Identity Management infrastructure are also created under the `cn=users`
    container.
  - All new users created in the identity management realm using the Oracle
    Identity Management infrastructure belong to the object classes `orclUserV2`
    and `inetOrgPerson`.
  - All groups are located in the container `cn=groups` under the base of the
    identity management realm. In this example, it is
    `cn=groups,dc=us,dc=mycompany,dc=com`.

- Identity management realm administrator. This user is located under the users
  container. In this example, the fully qualified DN of the realm administrator is
  `orcladmin, cn=users, dc=us, dc=mycompany, dc=com`.

---

**Note:** If the realm administrator’s account becomes locked, the
Oracle Internet Directory super user can unlock it by modifying
the realm administrator’s account password, using Oracle Directory
Manager.

---

- Default authentication policies, which enable authentication services to perform
the appropriate actions. These include:
  - The default directory password policy—for example, password length,
    lockout, and expiration
  - Additional password verifiers that need to be automatically generated when
    provisioning the user

- Identity management authorizations. Oracle Internet Directory grants these to the
realm administrator who can further delegate these authorizations through the
Oracle Internet Directory Self-Service Console. Some of these authorizations
include:
  - Common identity management operational privileges—for example, user
    creation, user profile modification, group creation
– Privileges to install new Oracle components by using the Oracle Identity Management infrastructure.
– Privileges to administer Oracle Internet Directory Self-Service Console

See Also:
- "Oracle Identity Management LDAP Object Classes" in Oracle Identity Management User Reference for more information about the orclUserV2 object class
- Chapter 21, "Delegation of Privileges for an Oracle Technology Deployment" for a fuller description of the default access control policies in Oracle Identity Management

Administration of Identity Management Realms
This section describes the various administrative tasks that you can perform with respect to identity management realms. It contains these topics:
- Customizing the Default Identity Management Realm
- Creating Additional Identity Management Realms for Hosted Deployments

Customizing the Default Identity Management Realm
Once a realm is created, you can further customize various aspects of it. Table 23–2 lists the aspects you can customize, the tools available for each type of customization, and where to look for more information.

Table 23–2 Customizing the Default Identity Management Realm

<table>
<thead>
<tr>
<th>What You Can Customize</th>
<th>Tools</th>
<th>Information</th>
</tr>
</thead>
</table>
| Directory structure and naming policies | Oracle Internet Directory Self-Service Console  
Oracle Directory Manager  
Command-line tools | "Changing the Location of Users and Groups In The Default Identity Management Realm" on page 23-13  
"Planning the Directory Information Tree for Identity Management" on page 23-1  
The chapter on using the Oracle Internet Directory Self-Service Console in Oracle Identity Management Guide to Delegated Administration |
| Authentication policies | Oracle Directory Manager  
Command-line tools | Chapter 19, "Password Policies in Oracle Internet Directory" |
| Identity management authorizations | Oracle Internet Directory Self-Service Console  
Oracle Directory Manager  
Command-line tools | Chapter 21, "Delegation of Privileges for an Oracle Technology Deployment"  
The chapter on using the Oracle Internet Directory Self-Service Console in Oracle Identity Management Guide to Delegated Administration |
Changing the Location of Users and Groups In The Default Identity Management Realm

A typical scenario where this might be required is one where you need to integrate your Oracle Identity Management installation with a third-party directory.

For example, assume the default Identity Management Realm is dc=mycompany,dc=com and there are users under cn=users,dc=mycompany,dc=com.

If the third party directory naming context does not match the current user and group search base in the default realm, then you can alter the user and group search base of the default realm so that both the existing users and the third party users can login using SSO. Select a user search base just high enough to include the existing users and the third party users. Let us call this search base the Lowest Common User Search Base.

Note: This approach assumes that the user nickname attribute selected for SSO Login is unique across the existing user search base and the third party directory naming context. Otherwise, SSO authentication will fail for all those users whose nickname attribute values clash.

If your deployment scenario matches any of the use cases from 1 to 5, follow the procedure described in "Steps to Update the Existing User and Group Search Base" on page 23-14.

Use Case 1:
The third party naming context is under the default realm, but in a different container than the realm user search base

For example, the existing users are under cn=users,dc=mycompany,dc=com and the third party naming context is under cn=users,o=employees,dc=mycompany,dc=com. In this case, the Lowest Common User Search Base is dc=mycompany,dc=com.

Use Case 2:
The third party naming context is outside the default realm, but there is a Lowest Common User Search Base.

For example, the existing users are under cn=users,dc=mycompany,dc=com and the third party naming context is under cn=users, o=mycompanyecorp,dc=com. In this case, the Lowest Common User Search Base is dc=com.

If the Lowest Common User Search Base is the root DSE, then use the procedures described for Use Case 6:

2. "Refresh SSO" on page 23-16.

Use Case 3:
The third party naming context is the same as the default realm DN.
For example, the existing users are under cn=users, dc=mycompany, dc=com and the third party naming context is directly under dc=mycompany, dc=com. In this case, the Lowest Common User Search Base is dc=mycompany, dc=com.

Use Case 4:
The third party naming context contains the parent of the default realm DN.

For example, you might have a default realm with DN: dc=us, dc=mycompany, dc=com, existing users under cn=users, dc=us, dc=mycompany, dc=com and the third party naming context directly under dc=com. In this case, the Lowest Common User Search Base is dc=com.

Use Case 5:
The third party naming context is under the existing user search base.

For example, existing users are under cn=users, dc=mycompany, dc=com and the third party naming context is directly under l=emea, cn=users, dc=mycompany, dc=com. In this case, the Lowest Common User Search Base is cn=users, dc=mycompany, dc=com. In this use case, you do not need to change the user search base.

Steps to Update the Existing User and Group Search Base

You must perform the following steps before you set up synchronization with the third party directory.

1. Back up the Oracle Internet Directory database.
2. Create the user and group containers for the third party directory, using Oracle Directory Manager, if the entries do not already exist in the directory.
3. Apply appropriate ACLs on the new users container by doing the following:
   a. Instantiate the variables %USERBASE% and %REALMBASE% in the ACL template file $ORACLE_HOME/ldap/schema/oid/oidUserAdminACL.sbs and create the file usracl.ldif. Set the variable %USERBASE% to the DN of the new user container and the variable %REALMBASE% to the default realm DN.
   b. Upload the instantiated LDIF file usracl.ldif using the ldapmodify command.
4. Apply appropriate ACLs on the new groups container by doing the following:
   a. Instantiate the variables %GRPBASE% and %REALMBASE% in the ACL template file $ORACLE_HOME/ldap/schema/oid/oidGroupAdminACL.sbs and create the file grpacl.ldif. Set the variable %USERBASE% to the DN of the new user container and the variable %REALMBASE% to the default realm DN.
   b. Upload the instantiated LDIF file grpacl.ldif using the ldapmodify command.
5. Determine a Lowest Common User Search Base base that is just high enough to include the existing users and the third party users.

For example, if existing users are under cn=users, dc=mycompany, dc=com and the third party users are under l=emea, dc=mycompany, dc=com, then the Lowest Common User Search Base is dc=mycompany, dc=com.

The Lowest Common User Search Base might be the root entry. That is the case if, for example, the existing users are under dc=mycompanycorp, dc=net. In that case,
skip to the deployment scenario described in Use Case 6: "Set up an Additional Search Base" on page 23-15

6. If you must also synchronize groups, determine a group search base that is just high enough to include the existing groups and the third party groups. Let's call this search base the Lowest Common Group Search Base.

For example, if existing groups are under `cn=groups,dc=mycompany,dc=com` and the third party groups are under `1=emea,dc=mycompany,dc=com`, then the Lowest Common Group Search Base is `dc=mycompany,dc=com`.

7. Log into the Self-Service Console as the administrator of the realm (usually `orcladmin`).

8. Go to the Configuration tab and set the user search base to the Lowest Common User Search Base you determined in step 5. If you must also synchronize groups, then also set the group search base to the Lowest Common Group Search Base that you determined in step 6.

9. To make SSO recognize these changes, follow the procedure described under "Refresh SSO" on page 23-16.

10. Verify the SSO login of users in the original user search base by logging in as `orcladmin`.

11. You must also reconfigure the applications that have been provisioned to reflect the modified user and group bases. Follow the steps described under "Reconfigure Provisioning Profiles" on page 23-17.

---

**Note:** In addition to the user and group search base attributes, you can also modify other configuration settings of an identity management realm, such as the attribute for Login Name (nickname) or the attribute for RDN, using the Self-Service Console. See: "Modifying Configuration Settings for an Identity Management Realm" in the chapter "Using the Oracle Internet Directory Self-Service Console" of Oracle Identity Management Guide to Delegated Administration for more details.

---

**Use Case 6:**

In this case, the third party naming context is outside the default realm and the Lowest Common User Search Base is the root DSE.

For example, if existing users are under `cn=users,dc=mycompany,dc=com` and the third party naming context is under `cn=users,dc=mycompanycorp,dc=net`, then the Lowest Common User Search Base is the root DSE.

In this case, you must add the third party naming context as an additional search base. The steps are as follows:

1. "Set up an Additional Search Base"
2. "Refresh SSO"
3. "Reconfigure Provisioning Profiles"

**Set up an Additional Search Base** Perform the following steps before setting up synchronization with the third party directory.

1. Back up the Oracle Internet Directory database.
2. Create the user and group containers for the third party directory, using Oracle Directory Manager, if the entries do not already exist in the directory.

3. Apply appropriate ACLs on the new users container by doing the following:
   a. Instantiate the variables %USERBASE% and %REALMBASE% in the ACL template file $ORACLE_HOME/ldap/schema/oid/oidUserAdminACL.sbs and create the file usracl.ldif. Set the variable %USERBASE% to the DN of the new user container and the variable %REALMBASE% to the default realm DN.
   b. Upload the instantiated LDIF file usracl.ldif using the ldapmodify command.

4. Apply appropriate ACLs on the new groups container by doing the following:
   a. Instantiate the variables %GRPBASE% and %REALMBASE% in the ACL template file $ORACLE_HOME/ldap/schema/oid/oidGroupAdminACL.sbs and create the file grpacl.ldif. Set the variable %USERBASE% to the DN of the new user container and the variable %REALMBASE% to the default realm DN.
   b. Upload the instantiated LDIF file grpacl.ldif using the ldapmodify command.

5. Log into the Self-Service Console as the administrator of the realm.

6. Go the Configuration tab.
   a. Add cn=users,dc=mycompanycorp,dc=net to the usersearchbase for the current realm.
   b. Add cn=groups,dc=mycompanycorp,dc=net to the groupsearchbase for the current realm.

7. To make Oracle Application Server Single Sign-On recognize these changes, follow the procedure described under "Refresh SSO" on page 23-16.

8. Verify the SSO login of users in the original user search base by logging in as orcladmin.

9. If mid-tiers have been configured against this identity management configuration, then you must also reconfigure the applications that have been provisioned to reflect the modified user and group bases. Follow the steps described under "Reconfigure Provisioning Profiles" on page 23-17.

---

**Note:** In addition to the user and group search base attributes, you can also modify other configuration settings of an identity management realm, such as the attribute for Login Name (nickname) or the attribute for RDN, using the Self-Service Console. See: "Modifying Configuration Settings for an Identity Management Realm" in the chapter "Using the Oracle Internet Directory Self-Service Console" of Oracle Identity Management Guide to Delegated Administration for more details.

---

**Refresh SSO** The steps for making Oracle Application Server Single Sign-On recognize your configuration changes are as follows:

1. Execute the SSO refresh script by changing to the directory $ORACLE_HOME/sso/admin/plsql/sso/ and typing:

   sqlplus orasso/password@ssoreoid.sql
To get the orasso schema password, refer to the appendix "Obtaining the Single Sign-On Schema Password" in the Oracle Application Server Single Sign-On Administrator's Guide.

2. Restart the OC4J_SECURITY instance by typing:

```bash
opmnctl restartproc type=oc4j instancename=OC4J_SECURITY
```

Reconfigure Provisioning Profiles  If you installed middle-tier applications against this default identity management realm before changing its user and group search bases, then the provisioning profiles created by the middle-tier installations become invalid. This happens because the profiles have the old user or group search base information in the event subscriptions attribute. You must modify all the profiles by using oidprovtool.

Execute the following steps for every provisioning profile:

1. Use ldapsearch to put all the provisioning profile information into an LDIF file:

```bash
ldapsearch -h oid_host -p oid_port -D "cn=orcladmin" -w password -s sub -b "cn=provisioning profiles,cn=changelog subscriber,cn=oracle internet directory" "objectclass=*" > provprofiles.ldif
```

The event subscriptions look something like this:

```
USER:cn=users,dc=mycompany,dc=com:MODIFY(list_of_attributes)
USER:cn=users,dc=mycompany,dc=com:DELETE
GROUP:cn=groups,dc=mycompany,dc=com:MODIFY(list_of_attributes)
GROUP:cn=groups,dc=mycompany,dc=com:DELETE
```

where `cn=users,dc=mycompany,dc=com` and `cn=groups,dc=mycompany,dc=com` are the user and group search bases, respectively, that were created when you installed and configured the application.

2. Get the actual DNs of the application identity by searching the Oracle Internet Directory server based on the GUID. To get the application DN, type:

```bash
ldapsearch -h host -p port -D cn=orcladmin -w password -s sub -b "" "orclguid=Value_of_orclODIPProvisioningAppGuid" dn
```

You can get the GUID values for each profile from the attribute values in provprofiles.ldif.

3. Modify each of the returned profiles as follows:

```bash
$ORACLE_HOME/bin/oidprovtool operation=MODIFY \
ldap_host=host ldap_port=port \
ldap_user_dn="cn=orcladmin" \ 
ldap_user_passwd=password \ 
interface_version=interfaceVersion \ 
application_dn=applicationDN \ 
organization_dn=identity_Realm_DN \ 
event_subscriptions=New_Event_Subscription_1 \ 
event_subscription=New_Event_Subscription_2
```

event_subscription=New_Event_Subscription

The `New_Event_Subscription` arguments should be of the form:

- USER: `new_user_search_base`: `MODIFY(list_of_attributes)`
- USER: `new_user_search_base`: `DELETE`
- GROUP: `new_group_search_base`: `MODIFY(list_of_attributes)`
- GROUP: `new_group_search_base`: `DELETE`

Here, the `organization_dn` value should be the original identity realm DN

### Creating Additional Identity Management Realms for Hosted Deployments

You can create additional identity management realms by using the Oracle Internet Directory Self-Service Console.

Only members of the ASPAdmins group can create a new identity management realm. Use Oracle Directory Manager to add a user to that group by adding the userDN to the uniquemember attribute of group ASPAdmins in the Default Identity Management Realm-specific OracleContext. Refer to the section on "Modifying a Static Group Entry by Using Oracle Directory Manager" for details.

**Note:** Not all applications can work with multiple identity management realms.

Whenever you add an additional realm, you may need to make existing applications aware of it by using a manual procedure. For more information, see the application-specific documentation.

In the Oracle Identity Management infrastructure, the single sign-on server needs to be made aware of an additional realm by using a special administrative procedure. Please refer to the chapter "Single Sign-On in Multiple Realms" in the Oracle Application Server Single Sign-On Administrator's Guide for instructions on enabling multiple realms in Oracle Application Server Single Sign-On.

**See Also:**

- The chapter on the Oracle Internet Directory Self-Service Console in the Oracle Identity Management Guide to Delegated Administration for information about creating an additional identity management realm.
Capacity Planning for the Directory

Capacity planning is the process of assessing applications’ directory access requirements and ensuring that the Oracle Internet Directory has adequate computer resources to service requests at an acceptable rate. This chapter explains what you need to consider when doing capacity planning. It guides you through an example of a directory deployment for an e-mail messaging application in a hypothetical company called Acme Corporation.

This chapter contains these topics:

- **About Capacity Planning**
- **Getting to Know Directory Usage Patterns: A Case Study**
- **I/O Subsystem Requirements**
- **Memory Requirements**
- **Network Requirements**
- **CPU Requirements**
- **Summary of Capacity Plan for Acme Corporation**

### About Capacity Planning

If Oracle Internet Directory and the corresponding Oracle Database are running on the same computer, then these are the configurable resources that capacity planners need to consider:

- I/O subsystem (the type and size)
- Memory
- Network connectivity
- CPUs (speed and quantity)

When you plan to acquire hardware for Oracle Internet Directory, you should ensure that all components—such as CPU, memory, and I/O—are effectively used. Generally, good memory usage and a robust I/O subsystem are sufficient to keep the CPU busy.

To be successful, every new installation of the Oracle Internet Directory requires:

- Adequate hardware resources to satisfy user demands at peak load rates
- A well tuned system—of both hardware and software—that makes the best use of available resources and squeezes the maximum performance out of available hardware

Table 24–1 defines important terms used in this chapter.
Getting to Know Directory Usage Patterns: A Case Study

The ability to assess the potential load on Oracle Internet Directory is very important for developing an accurate capacity plan. Let us examine the e-mail messaging software employed by our hypothetical company, Acme Corporation. The e-mail messaging software in this example is based on Internet Message Access Protocol (IMAP). There are two main types of software that access Oracle Internet Directory:

- The IMAP clients, which will validate e-mail addresses within the company before sending the mail to the IMAP server. These clients include software programs like Netscape Messenger and Microsoft Outlook.
- The messaging software itself, also called the Mail Transfer Agent (MTA), which will look up the directory to route mail from the outside world to internal mailboxes as well as route internal mails to company-wide distribution lists.

Let us assume that the private aliases and private distribution lists of individual users are also stored in the directory. Let us further make the assumptions in Table 24–2 that enable us to guess the size of the directory.

### Table 24–1 Capacity Planning Terminology

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Throughput</td>
<td>The overall rate at which directory operations are being completed by Oracle Internet Directory. This is typically represented as “operations every second.”</td>
</tr>
<tr>
<td>Latency</td>
<td>The time a client has to wait for a given directory operation to complete</td>
</tr>
<tr>
<td>Concurrent clients</td>
<td>The total number of clients that have established a session with Oracle Internet Directory</td>
</tr>
<tr>
<td>Concurrent operations</td>
<td>The amount of concurrent 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.</td>
</tr>
</tbody>
</table>

In this chapter, we look at an example of a directory deployment for an e-mail messaging application in a hypothetical company called Acme Corporation. As we examine each component of the capacity plan, we apply our recommendations to the example of Acme Corporation.

### Table 24–2 Assumptions about Entry Types and Their Sizes

<table>
<thead>
<tr>
<th>Entry Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total user population</td>
<td>40,000</td>
</tr>
<tr>
<td>Average number of private aliases for each person</td>
<td>10</td>
</tr>
<tr>
<td>Average number of private distribution lists for each person</td>
<td>10</td>
</tr>
<tr>
<td>Total number of public distribution lists</td>
<td>4000</td>
</tr>
<tr>
<td>Total number of public aliases in the company</td>
<td>1000</td>
</tr>
<tr>
<td>Number of attributes in each entry in the directory related to this application</td>
<td>20</td>
</tr>
<tr>
<td>Number of cataloged attributes</td>
<td>10</td>
</tr>
</tbody>
</table>
Based on these assumptions, we can derive the overall count of entries in Oracle Internet Directory as described in Table 24–3.

<table>
<thead>
<tr>
<th>Entry Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>User entries</td>
<td>40,000 (these represent the users themselves)</td>
</tr>
<tr>
<td>Private aliases of users</td>
<td>40,000 x 10 = 400,000 entries</td>
</tr>
<tr>
<td>Private distribution lists of users</td>
<td>40,000 x 10 = 400,000 entries</td>
</tr>
<tr>
<td>Company wide distribution lists</td>
<td>4000</td>
</tr>
<tr>
<td>Company wide aliases</td>
<td>1000</td>
</tr>
</tbody>
</table>

These assumptions yield a directory population of approximately one million entries. Given the user population and the directory population, let us then analyze usage patterns so that we can derive performance requirements from them. A typical user tends to send an average of 10 e-mails everyday and receives an average of 10 e-mails a day from the outside world. Assuming an average of five recipients for each e-mail sent by a user, this would result in five directory lookups for each e-mail.

Table 24–4 summarizes all the possible directory lookups that can happen in one day.

<table>
<thead>
<tr>
<th>Type of Directory Lookup</th>
<th>Number of Directory Lookups In One Day</th>
</tr>
</thead>
<tbody>
<tr>
<td>The Mail Transfer Agent (MTA) processing outbound mail from each user</td>
<td>5x10x40,000 = 2,000,000</td>
</tr>
<tr>
<td>The MTA processing mails from the outside world</td>
<td>10x40,000 = 400,000</td>
</tr>
<tr>
<td>All other directory lookups (like IMAP clients validating certain addresses, and so on)</td>
<td>800,000</td>
</tr>
</tbody>
</table>

To summarize: The total number of directory lookups everyday would be about 3,200,000 (3.2 million). If these lookups were spread out uniformly throughout the day, it would require about 37 directory lookups every second (133,333 lookups every hour). Unfortunately, we will never have this case.

Usage analysis of the current e-mail system over a period of 24 hours shows the pattern illustrated in Figure 24–1.
The e-mail system and Oracle Internet Directory are maximally stressed in the mornings. There are other usage peaks as well: one close to lunch time, and one near the end of business day. However, it is in the mornings that the Oracle Internet Directory is stressed the most.

Let us assume that 90 percent of all the directory lookups happen during normal working hours. Table 24–5 shows the shift load for the morning, afternoon, and evening periods of an eight hour day.

<table>
<thead>
<tr>
<th>Shift Load</th>
<th>Lookups</th>
</tr>
</thead>
<tbody>
<tr>
<td>Morning load</td>
<td>65%: $0.90 \times 0.65 \times 3,200,000 = 1,872,000$ lookups for 2 hours (936,000 lookups every hour)</td>
</tr>
<tr>
<td>Afternoon load</td>
<td>10%: $0.90 \times 0.10 \times 3,200,000 = 288,000$ lookups for 1 hour (288,000 lookups every hour)</td>
</tr>
<tr>
<td>Evening load</td>
<td>20%: $0.90 \times 0.20 \times 3,200,000 = 576,000$ lookups for 2 hours (288,000 lookups every hour)</td>
</tr>
</tbody>
</table>

These calculations indicate that the directory in this case should be designed to handle the peak load of 936,000 lookups every hour.

Now that we know the data-set size as well as the performance requirements, we can look into individual components of the installation and estimate good values for each.

I/O Subsystem Requirements

This section contains these topics:

- About the I/O Subsystem
- Rough Estimates of Disk Space Requirements
- Detailed Calculations of Disk Space Requirements
About the I/O Subsystem

The I/O subsystem can be compared to a pump that sends data to the CPUs to enable them to execute workloads. The I/O subsystem is also responsible for data storage. The main components of an I/O subsystem are arrays of disk drives controlled by disk controllers.

It is important to consider performance requirements when you size the I/O subsystem, rather than size based only on storage requirements. Although disk drives have increased in size, the throughput—that is, the rate at which the disk drive pumps data—has not increased in proportion. In sizing calculations for the I/O subsystem, you should use the following factors as input:

- The size of the database
- The number of CPUs on the system
- An initial estimation of the workload on the Oracle Internet Directory
- The rate at which the disk can pump data
- Space needed to stage data prior to load
- Space needed for index creation and sort activities

Given a range of I/O subsystems, you should always opt for the highest throughput drives. Typically, one can maximize the I/O throughput by one or more of the following techniques:

- Striping logical volumes so that the I/O operations use multiple disk spindles
- Putting different tablespaces in different logical and physical disk volumes
- Distributing the disk volumes on multiple I/O controllers

Some guidelines for organizing Oracle Internet Directory-specific data files are provided in Chapter 25, "Tuning Considerations for the Directory". Depending on the tolerance of disk failures, different levels of Redundant Arrays of Inexpensive Disks (RAID) can also be considered.

Assuming that the decision has been made to get the best possible I/O subsystem, we focus the next section on deriving sizing estimates for the disks themselves.

Rough Estimates of Disk Space Requirements

You can use Table 24–6 to derive a rough estimate of the overall disk requirement.

<table>
<thead>
<tr>
<th>Number of Entries in DIT</th>
<th>Disk Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>100,000</td>
<td>450MB to 650MB</td>
</tr>
<tr>
<td>200,000</td>
<td>850MB to 1.5GB</td>
</tr>
<tr>
<td>500,000</td>
<td>2.5GB to 3.5GB</td>
</tr>
<tr>
<td>1,000,000</td>
<td>4.5GB to 6.5GB</td>
</tr>
<tr>
<td>1,500,000</td>
<td>6.5GB to 10GB</td>
</tr>
<tr>
<td>2,000,000</td>
<td>9GB to 13GB</td>
</tr>
</tbody>
</table>

The data shown in Table 24–6 makes the following assumptions:

- There are about 20 cataloged attributes.
There are about 25 attributes for each entry.

The average size of an attribute is about 30 bytes.

Going back to our example of Acme Corporation, since our directory population is about one million, this would imply that our disk requirements are approximately 4.5 GB to 6.5 GB. Note that the assumptions made for Acme Corporation regarding the number of cataloged attributes are different, but the previous table should give an approximate figure of the size requirements.

Since the directory may be deployed for a wide variety of applications, these assumptions need not necessarily hold true for all possible situations: There might be cases where the size of attributes is large, the number of attributes for each entry is large, extensive use of ACIs has been made, or the number of cataloged attributes is very high. For such cases, we present simple arithmetic procedures in the following section which will allow the planners to get a more detailed perspective of their disk requirements.

Detailed Calculations of Disk Space Requirements

Because Oracle Internet Directory stores all of its data in an Oracle Database, the sizing for disk space is primarily a sizing of the underlying database. Oracle Internet Directory stores its data in the tablespaces described in Table 24–7.

Table 24–7 Tablespace Used to Store Oracle Internet Directory Data

<table>
<thead>
<tr>
<th>Tablespace Name</th>
<th>Size</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>OLTS_ATTR_STORE</td>
<td>3.5 times LDIF file size</td>
<td>Stores all of the directory data</td>
</tr>
<tr>
<td>OLTS_BATTRSTORE</td>
<td>1.5 times binary data values size</td>
<td>Stores all binary data. If no binary data is being loaded, leave this at the default size.</td>
</tr>
<tr>
<td>OLTS_CT_STORE</td>
<td>3.5 times LDIF file size</td>
<td>Stores catalog tables for fast lookups. Size depends on the number of cataloged attributes. The estimate is based on the default configuration.</td>
</tr>
<tr>
<td>OLTS_DEFAULT</td>
<td>See comments.</td>
<td>Used during the -check phase of bulkload and for holding change log data and some replication and process configuration tables. The size should be largely based on the number of updates on the directory and the frequency of the purge configuration.</td>
</tr>
<tr>
<td>OLTS_SVRMGSTORE</td>
<td>2 M</td>
<td>Used for Oracle Internet Directory Server statistics when statistics are configured. In most cases, the default is adequate.</td>
</tr>
</tbody>
</table>

This section presents simple arithmetic procedures to determine the size requirements of each of the tablespaces referred to in Table 24–7. All of the size calculations are based on the variables in Table 24–8.

Table 24–8 Variables Used for Size Calculation

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>num_entries</td>
<td>Total number of entries in the directory</td>
</tr>
<tr>
<td>attrs_per_entry</td>
<td>Average number of attributes for each directory entry</td>
</tr>
<tr>
<td>avg_attr_size</td>
<td>Average size of the attribute value in bytes</td>
</tr>
<tr>
<td>avg_dn_size</td>
<td>Average size of the DN of an attribute in bytes</td>
</tr>
<tr>
<td>objectclass_per_entry</td>
<td>Average number of object classes that an entry belongs to</td>
</tr>
</tbody>
</table>
Using the variables shown in Table 24–8, the size of individual tablespaces can be calculated as shown in Table 24–9.
Use the arithmetic operations shown in the preceding table to compute the exact space requirements for a wide variety of Oracle Internet Directory deployment scenarios. The sum of the sizes of each of the tablespaces should yield the overall database disk requirement. One can optionally multiply that by the "factor_of_safety" variable to get a figure that can compensate for unforeseen circumstances.

Going back to our example of Acme Corporation, we can assign values to each of the variables based on the requirements stated in previous sections. Table 24–10 illustrates the values of each variable introduced in this section for Acme Corporation.

<table>
<thead>
<tr>
<th>Tablespaces Containing Tables</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>OLTS_BATTRSTORE</td>
<td>$6M+((\text{num_binary_attrs} \times \text{avg_binval_length}) + 6 \times 35) \times \text{db_storage_ovhd})$</td>
</tr>
<tr>
<td>OLTS_CT_STORE</td>
<td>$(\text{cn_size} + \text{objectclasses_size} + \text{dn_size} + \text{catalog_index_size} + \text{dn_index_size})$</td>
</tr>
<tr>
<td>OLTS_DEFAULT</td>
<td>$(\text{change_log_capacity} \times 4 \times \text{avg_attr_size} \times \text{db_storage_ovhd} \times \text{db_index_ovhd}) + (\text{initial_num_entries} \times 2 \times (\text{avg_dn_size} + 4))$</td>
</tr>
<tr>
<td>OLTS_SVRMGSTORE</td>
<td>$2M + \text{num_stats_entries} \times ((\text{avg_attrname_len} + \text{avg_attr_size} + 20) \times (2 \times \text{attrs_per_stats_entry}) \times \text{db_storage_ovhd} \times (\text{orclstatsperiodicity}/10) \times 12)$</td>
</tr>
<tr>
<td>SYSTEM</td>
<td>$300MB$</td>
</tr>
</tbody>
</table>

Table 24–10 Values for Variables Used for Sizing Calculations

<table>
<thead>
<tr>
<th>Variable Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>num_entries</td>
<td>1,000,000</td>
</tr>
<tr>
<td>attrs_per_entry</td>
<td>20</td>
</tr>
<tr>
<td>avg_attr_size</td>
<td>32 bytes</td>
</tr>
<tr>
<td>avg_dn_size</td>
<td>40 bytes</td>
</tr>
<tr>
<td>objectclass_per_entry</td>
<td>5 (each entry belongs to an average of 5 object classes)</td>
</tr>
<tr>
<td>objectclass_size</td>
<td>10 bytes</td>
</tr>
<tr>
<td>num_cataloged_attrs</td>
<td>10</td>
</tr>
<tr>
<td>entries_per_catalog</td>
<td>1,000,000</td>
</tr>
<tr>
<td>change_log_capacity</td>
<td>80,000 changes (2 for each user)</td>
</tr>
<tr>
<td>num_acis</td>
<td>80,000 ACIs (2 for each user)</td>
</tr>
<tr>
<td>num_auditlog_entries</td>
<td>1000</td>
</tr>
<tr>
<td>db_storage_ovhd</td>
<td>1.4 (40% overhead)</td>
</tr>
<tr>
<td>db_index_ovhd</td>
<td>5.0 (400% overhead)</td>
</tr>
<tr>
<td>factor_of_safety</td>
<td>1.5 (50% factor of safety)</td>
</tr>
<tr>
<td>initial_num_entries</td>
<td>1,000,000</td>
</tr>
<tr>
<td>num_stats_entries</td>
<td>5</td>
</tr>
<tr>
<td>attrs_per_stats_entry</td>
<td>12</td>
</tr>
<tr>
<td>orclstatsperiodicity</td>
<td>60 (root DSE attribute)</td>
</tr>
<tr>
<td>avg_attrname_len</td>
<td>6</td>
</tr>
</tbody>
</table>
If we now plug these values into the equations described earlier, we get the values listed in Table 24–11.

**Table 24–11 Tablespace Sizes**

<table>
<thead>
<tr>
<th>Tablespace Name</th>
<th>Size in Bytes</th>
<th>Size in MB</th>
</tr>
</thead>
<tbody>
<tr>
<td>OLTS_ATTRSTORE</td>
<td>2,223,000,000</td>
<td>2182</td>
</tr>
<tr>
<td>OLTS_CT_STORE</td>
<td>2,328,512,000</td>
<td>274</td>
</tr>
<tr>
<td>OLTS_DEFAULT</td>
<td>159,680,000</td>
<td>156</td>
</tr>
<tr>
<td>OLTS_SRVMGSTORE</td>
<td>2,701,568</td>
<td>3</td>
</tr>
<tr>
<td>SYSTEM</td>
<td>314572800</td>
<td>300</td>
</tr>
<tr>
<td><strong>Total Size</strong></td>
<td><strong>5038093862</strong></td>
<td><strong>4920</strong></td>
</tr>
</tbody>
</table>

Table 24–11 shows that the estimated size of the database for Acme Corporation would be about 8.25 GB. If all of the data is being loaded in bulk, then the bulkload tool of Oracle Internet Directory would require an additional 30 percent of space occupied by the database to store its temporary files. For Acme Corporation, this would add about 2.5 GB to the total space requirement.

### Memory Requirements

Memory is used for a number of distinct tasks by any database application, including Oracle Internet Directory. If memory resources are insufficient for any of these tasks, then the CPUs work less efficiently and system performance drops. Furthermore, memory usage increases in proportion to the number of concurrent connections to the database and the number of concurrent users of the directory. For the purposes of capacity planning, an active connection begins when a client seeks to bind to the directory and ends when that bind is terminated.

The memory available to processes comes from the virtual memory on the system, which is somewhat more than available physical memory. If the sum of all active memory usage exceeds the available physical memory on the system, the operating system may need to store some of the memory pages on disk. This is called paging. Paging can degrade performance if memory is too oversubscribed. Generally, you should not exceed 20 percent over-subscription of physical memory. If paging occurs, you need either to scale back memory usage by processes or to add more physical memory. Keep in mind the trade-offs: There are physical limits to the amount of memory you can add, but scaling back on memory usage for each process can significantly degrade performance.

The main consumers of memory are the database buffer cache within the system global area (SGA) and the Oracle Internet Directory Server Entry Cache (if enabled). Getting a good hit ratio for the buffer cache and the entry cache requires allocating enough memory in each area. The following formula gives a rough estimate for the amount of RAM required to cache 'N' entries in the entry cache:

\[
N \times [150 + \text{attrs_per_entry} + 6] \times (\text{avg_attrname_len} + \text{avg_attr_size} + 40) \times 1.3
\]

**See Also:** Chapter 25, "Tuning Considerations for the Directory" for further information on SGA tuning.

Table 24–12 gives minimum memory requirements for different directory configurations.
Going back to our example of Acme Corporation, the number of entries in the directory are close to 1,000,000 (1 million). Oracle Corporation recommends choosing the 2 GB option in order to maximize performance.

**Network Requirements**

The network is rarely a bottleneck in most installations. However serious consideration must be given to it during the capacity planning stage. If the clients do not get adequate network bandwidth to send and receive messages from Oracle Internet Directory, the overall throughput will seem to be very low. For example, if we have configured Oracle Internet Directory to service 800 search operations every second, but the computer running the Oracle directory server is only accessible through a 10 Mbps network (10-Base-T switched ethernet), and we have only 60 percent of the bandwidth available, then the clients will only see a throughput of 60 search operations a second (assuming each search operation causes 1024 bytes to be transferred on the network). Table 24–13 shows the maximum possible throughput (in operations every second) for two types of operations (one requiring a transfer of 1024 bytes the other requiring a transfer of 2048 bytes) for two types of networks, 10 Mbps & 100 Mbps, at different rates of bandwidth availability.

<table>
<thead>
<tr>
<th>Percent Available Bandwidth</th>
<th>1024 bytes Operations/sec</th>
<th>2048 bytes Operations/sec</th>
</tr>
</thead>
<tbody>
<tr>
<td>10 Mbps</td>
<td>300</td>
<td>300</td>
</tr>
<tr>
<td>100 Mbps</td>
<td>3000</td>
<td>3000</td>
</tr>
<tr>
<td>150</td>
<td>1500</td>
<td>1500</td>
</tr>
<tr>
<td>200</td>
<td>2000</td>
<td>2000</td>
</tr>
<tr>
<td>250</td>
<td>2250</td>
<td>2250</td>
</tr>
<tr>
<td>300</td>
<td>2500</td>
<td>2500</td>
</tr>
<tr>
<td>30</td>
<td>300</td>
<td>300</td>
</tr>
<tr>
<td>40</td>
<td>400</td>
<td>400</td>
</tr>
<tr>
<td>50</td>
<td>500</td>
<td>500</td>
</tr>
<tr>
<td>60</td>
<td>600</td>
<td>600</td>
</tr>
<tr>
<td>70</td>
<td>700</td>
<td>700</td>
</tr>
<tr>
<td>80</td>
<td>800</td>
<td>800</td>
</tr>
<tr>
<td>90</td>
<td>900</td>
<td>900</td>
</tr>
</tbody>
</table>

In some cases, it may also be important to consider the network latency of sending a message from a client to the Oracle directory server. In some WAN implementations, the network latencies may become as high as 500 milliseconds, which may cause the clients to time out for certain operations. In summary, given a range of networking options, the preferred choice should always be for highest bandwidth, lowest latency network.

Going back to the example of Acme Corporation, their peak usage rate is 936,000 lookups every hour which results in an equivalent number of lookup operations to the directory. This requires about 260 directory operations every second. Assuming that each operation results in a transfer of 2 KB of data on the network, this would imply...
that we should have a 100 Mbps network or at least 60 percent bandwidth available on
a 10 Mbps network. Since the 100 Mbps network will typically have a lower latency,
we will chose that over the 10 Mbps network.

CPU Requirements

This section contains these topics:

- CPU Configuration
- Rough Estimates of CPU Requirements
- Detailed Calculations of CPU Requirements

CPU Configuration

The CPU sizing for Oracle Internet Directory is directly a function of the user
workload. The following factors will determine CPU configuration:

- The number of concurrent operations you want to support. This will be directly
dependent on the number of users performing operations simultaneously.
- The acceptable latency of each operation. For example, in an e-mail application, a
  latency for each operation of 100 milliseconds might be desirable, but in most
cases a latency of 500 milliseconds might still be acceptable.

CPU resources can be added to a system as the workload increases, but these additions
seldom bring linear scalability to all operations since a lot of operations are not purely
CPU bound. We classify the processing power of a computer by a performance
characteristic that is commonly available from all vendors, namely, SPECint_rate95
baseline. This number is derived from a set of integer tests and is available from all
system vendors as well as the SPEC Web site (http://www.spec.org).

Note: SPECint_rate95 should not be confused with the regular
SPECint95 performance number. The SPECint95 performance
number gives an idea of the integer processing power of a
particular CPU (for systems with multiple CPUs, this number is
typically normalized). The SPECint_rate95 gives the integer
processing power of an entire system without any normalization.

Because Oracle Internet Directory makes efficient use of multiple CPUs on an SMP
computer, we chose to categorize computers based on their SPECint_rate95 numbers.
Even within SPECint_rate95 we chose the baseline number as opposed to the
commonly advertised result. This is because the commonly advertised result is
actually the peak performance of a computer, whereas the baseline number represents
the performance in normal circumstances.

Rough Estimates of CPU Requirements

Since Oracle Internet Directory typically co-resides with the Oracle Database, we
recommend at least a two-CPU system. We give the rough estimates in Table 24–14
based on the level of usage of Oracle Internet Directory.
Detailed Calculations of CPU Requirements

It is difficult to determine the CPU requirements for all operations at a given deployment site since the amount of CPU consumed depends upon several factors, such as:

- The type operation: base search, subtree search, modify, add, and so on
- If SSL mode is enabled or not, since SSL consumes an additional 15 to 20 percent of CPU resources.
- If Oracle Internet Directory server entry cache is enabled or not, since the hit ratio affects CPU usage.
- The number of entries returned for a search
- The number of access control policies that need to be checked as part of a search

In most of the cases, except SSL, we can expect that there is a large latency between the Oracle Internet Directory server process and the database. When a thread in the Oracle Internet Directory server process is waiting for the database to respond, other threads within the Oracle Internet Directory server process can be put to work by other client requests needing LDAP server specific processing. As a result, for any mix of operations, one can always come up with a combination of concurrent clients and Oracle Internet Directory server processes that will result in 100 percent CPU utilization. In this case, the CPU becomes the bottleneck.

Given this fact, we have taken a 'messaging' type of subtree search operation and tried to estimate the CPU resources need to support a given number of concurrent operations without degrading the throughput of operations. The 'messaging' search operation involves subtree scope, a simple exact match filter and a result set of one entry. For Oracle Internet Directory 10g (10.1.4.0.1):

\[
\text{SPECint}_\text{rate95} \text{ baseline} = 0.5 \times (\text{max # of concurrent operations at peak throughput})
\]

This means that, if we need to support 600 concurrent clients without degrading the throughput of operations, then we need a computer that has at least a SPECint_rate95 baseline rating of \((0.5 \times 600) = 300\).

In terms of throughput of operations, for Oracle Internet Directory 10g (10.1.4.0.1):

\[
\text{SPECint}_\text{rate95} \text{ baseline} = 0.4 \times (\text{throughput of operations at max supported concurrency})
\]

What this means is that if we need a throughput of 750 operations every second for the given maximum number of supported concurrent operations, then we need a computer that has at least a SPECint_rate95 baseline rating of \((0.4 \times 750) = 300\).

It has been proven that Oracle Internet Directory scales very well with additional CPU resources. What this means is:

- For a given concurrency of operations, we can achieve higher throughput of operations (and hence, a lower latency) by adding additional CPU resources.

### Table 24–14 Rough Estimates of CPU Requirements

<table>
<thead>
<tr>
<th>Usage</th>
<th>Num CPUs</th>
<th>SPECint_rate95 baseline</th>
<th>System</th>
</tr>
</thead>
<tbody>
<tr>
<td>Departmental</td>
<td>2</td>
<td>60 to 200</td>
<td>Compaq AlphaServer 8400 5/300 (300Mhz x 2)</td>
</tr>
<tr>
<td>Organization wide</td>
<td>4</td>
<td>200 to 350</td>
<td>IBM RS/6000 J50 (200MHz x 4)</td>
</tr>
<tr>
<td>Enterprise wide</td>
<td>4+</td>
<td>350+</td>
<td>Sun Ultra 450 (296 MHz x 4)</td>
</tr>
</tbody>
</table>
For a given throughput of operations (and latency), we can support higher concurrency of operations by adding additional CPU resources.

Going back to our example of Acme Corporation, let us assume that we want adequate CPU resources to support 500 concurrent ‘messaging’ type of subtree search operations with each client seeing subsecond latency. Taking a factor of safety of 20 percent, our preliminary estimate of CPU requirement would be a computer with a SPECint_rate95 baseline of at least 360.

Summary of Capacity Plan for Acme Corporation

In the preceding sections, we have described various components involved in capacity planning and have also shown how each of them would apply to an Oracle Internet Directory deployment at a hypothetical company named Acme Corporation. In this section we give a quick summary of all of the recommendations made. Following were the initial assumptions:

- Overall directory size: 3,200,000 entries (3.2 million)
- Number of users: 40,000
- Type of application: IMAP messaging
- Peak search rate: 750 searches/sec at concurrency of 500 clients

Based on these requirements and further assumptions, we developed the following recommendations:

- Disk space: 5 GB to 8 GB
- Memory: 2 GB
- Network: 100 Base-T
- CPU: something that has a SPECint_rate95 of at least 360.

Several simplifying assumptions were made so that the sizing calculations could be more intuitive.
Once you have completed capacity planning as described in Chapter 24, "Capacity Planning for the Directory", and you have acquired the necessary hardware, then you must ensure that the combined hardware and software are yielding the desired levels of performance. This chapter gives guidelines for tuning an Oracle Internet Directory installation. It contains these topics:

- About Tuning
- Tools for Performance Tuning
- CPU Usage Tuning
- Memory Tuning
- Disk Tuning
- Database Tuning
- Entry Caching
- Optimizing Searches
- Setting the Time Limit Mode
- Setting the Timeout for Client/Server Connections

See Also: "Troubleshooting Directory Performance" on page L-8

About Tuning

The two main performance metrics for any installation of Oracle Internet Directory are:

- The average latency of individual operations at peak load
  This is the time for each operation to complete.
- The overall throughput of Oracle Internet Directory expressed in operations for each second at peak load
  This is the rate at which an instance of Oracle Internet Directory is capable of completing client operations

If the performance tests yield poor results, the performance problems may be identified and fixed using the information provided in the following sections.
Tools for Performance Tuning

Knowledge of the following tools is recommended for Solaris and most other UNIX operating systems:

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>top</td>
<td>Displays the top CPU consumers on a system</td>
</tr>
<tr>
<td>vmstat</td>
<td>Shows running statistics on various parts of the system including the Virtual Memory Manager</td>
</tr>
<tr>
<td>mpstat</td>
<td>Shows an output similar to vmstat but split across various CPUs in the system. This is available on Solaris only.</td>
</tr>
<tr>
<td>iostat</td>
<td>Shows the disk I/O statistics from various disk controllers</td>
</tr>
</tbody>
</table>

Knowledge of the following tools is recommended for Microsoft Windows:

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows Performance Monitor</td>
<td>Provides a customized view of the events in the system</td>
</tr>
<tr>
<td>Windows Task Manager</td>
<td>Provides a high level output (like <code>top</code> on UNIX) of the major things happening in the system.</td>
</tr>
</tbody>
</table>

Knowledge of the following tools is recommended for the Oracle Database:

- `utlbstat.sql` and `utlestat.sql`, or `statspack`
- The ANALYZE function in the `DBMS_STATS` package

**See Also:**
- *Oracle Database Reference* in the Oracle Database Documentation Library for information about `utlbstat.sql` and `utlestat.sql`
- *Oracle Database Performance Tuning Guide* for information about `statspack`
- *Oracle Database Concepts* in the Oracle Database Documentation Library for information about the ANALYZE function in the `DBMS_STATS` package

In addition to the operating system tools, the LDAP applications being used in a customer environment must be able to provide latency and throughput measurement.

In addition, the Database Statistics Collection Tool (`oidstats.sql`), located at `$ORACLE_HOME/ldap/admin`, is provided to analyze the various database 'ods' schema objects to estimate the statistics.

**See Also:** The `oidstats.sql` command-line tool reference in *Oracle Identity Management User Reference*

CPU Usage Tuning

The CPU is perhaps the most important resource available for any software. While Chapter 24, "Capacity Planning for the Directory" gives a rough estimate of the required CPU horsepower for a given application load, sometimes insufficient tuning
can cause inefficient use of the CPU resources. Consider tuning CPU resources if either of the following cases is true:

- At peak loads the CPU is 100 percent utilized.
- At peak loads the CPU is under utilized, there is a significant amount of idle time in the system, and this idle time cannot be eliminated at even higher loads.

Internal benchmarks show that Oracle Internet Directory performs best when approximately 70 to 75 percent of the CPU resources are consumed by Oracle Internet Directory processes, and the remaining (about 25 to 30 percent) are consumed by the Oracle foreground processes corresponding to the database connections. While monitoring CPU usage, it is also important to monitor the percentage of time spent in the system space compared to user space. Internal benchmarks show best throughput numbers at about 85 percent user and 15 percent system time.

This section contains these topics:

- Tuning CPU for Oracle Internet Directory Processes
- Tuning CPU for Oracle Foreground Processes
- Taking Advantage of Processor Affinity on SMP Systems
- Other Alternatives for a CPU Constrained System

## Tuning CPU for Oracle Internet Directory Processes

The demands placed by Oracle Internet Directory processes on the CPU can be controlled by the `ORCLSERVERPROCS` and `ORCLMAXCC` parameters. Table 25–1 lists suggested values for these parameters for various client loads.

<table>
<thead>
<tr>
<th>ORCLSERVERPROCS</th>
<th>ORCLMAXCC</th>
<th># Concurrent clients supported without degrading throughput of operations</th>
<th># Clients supported without dropping connections</th>
<th>Required # of CPUs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>40</td>
<td></td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>400</td>
<td>800</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td>800</td>
<td>1600</td>
<td>4</td>
</tr>
<tr>
<td>8</td>
<td>10</td>
<td>1600</td>
<td>3200</td>
<td>8</td>
</tr>
</tbody>
</table>

If we take the example of 500 concurrent clients, a value of 4 for `ORCLSERVERPROCS` with a value of 10 for `ORCLMAXCC` will result in the following configuration:

- There will be four server processes created.
- Each server process will spawn 10 worker threads that will do the actual work.
- Each server process will also maintain a pool of sixteen database connections (10+5+1) that will be shared among the worker threads.

Oracle Internet Directory scales very well with CPU resources both with respect to the throughput of operations and concurrency of clients. From the previous table, say we have a 4 CPU box and are able to maintain a peak throughput of ‘p’ operations every second for a concurrency of ‘n’ clients.

With additional number of CPUs or with faster CPUs, we can achieve either or both of the following benefits:
CPU Usage Tuning

- Achieve a throughput higher than ‘p’ for the same concurrency of ‘n’ clients
- Maintain the same ‘p’ operations throughput for a concurrency higher than ‘n’

If the CPU usage at peak loads is not at 100 percent and the system is idle for a large percentage of the time (that is, more than 5 percent), this indicates that Oracle Internet Directory processes are under-configured and are not making the best utilization of the CPU resources. To solve this problem, one must systematically increase the values of ORCLSERVERPROCS and ORCLMAXCC until the CPU utilization reaches 100 percent and the system and user time are split up as follows:

- User time: 85 percent or higher
- System time: 15 percent or lower

Tuning CPU for Oracle Foreground Processes

Tuning of CPU resources for Oracle Foreground processes should be considered only if both of the following conditions are met:

- The CPU usage is close to 100 percent at peak loads.
- Oracle foreground processes consume more than 30 percent of all available CPU resources.

If Oracle foreground processes are consuming excessive CPU, it implies that the queries that Oracle Internet Directory is making against the database are using too many CPU cycles. Although there is very little control available to the users on the types of underlying operations performed by the database, the following should be attempted:

- Database statistics on all of the tables and indexes associated with the ODS user on the database must be collected using the `ANALYZE` command. This helps the cost-based optimizer make better execution plans for the queries generated by Oracle Internet Directory. `ORACLE_HOME/ldap/admin/oidstats.sql` can be used to collect statistics.
- If the `ANALYZE` fails to produce better results, and the LDAP queries used have a lot of filters in them, then a simple reorganization of the order in which the filters are specified (with the most specific filter in the beginning and the most generic filter at the end) helps reduce the CPU consumption of the Oracle foreground processes.

Taking Advantage of Processor Affinity on SMP Systems

Several Symmetric Multi-Processor (SMP) systems offer the capability to bind a particular process to a particular CPU. While it is generally a good idea not to bind any process to any processor, it may improve performance if the following conditions are met:

- The CPU utilization of the entire system is close to 100 percent.
- There are more than two CPUs on the computer.

In internal benchmarks, it has been observed that binding the Oracle Internet Directory Server process and its associated Oracle shadow processes to the same CPU generally gives the best performance.
Other Alternatives for a CPU Constrained System

If none of the tips stated in the preceding sections solve CPU related performance problems, the following options are available:

- Upgrade the processing power of the computer, that is, add more CPUs or replace slower CPUs with faster ones.
- Keep the Oracle directory server and the associated Oracle Database on separate computers.

Memory Tuning

After the CPU, memory is the next most important thing to tune. The primary consumer of memory in an Oracle Internet Directory installation is the Oracle Database. Make the SGA of the back-end database large enough while leaving room for Oracle Internet Directory and Oracle processes to operate their private stacks and heaps. This section provides some details on determining various components of the SGA.

This section contains these topics:

- Tuning the System Global Area (SGA) for the Oracle Database
- Other Alternatives for a Memory-Constrained System

Tuning the System Global Area (SGA) for the Oracle Database

The SGA should be sized based on the available physical memory on the system running the Oracle Database.

See Also: Oracle Database Performance Tuning Guide in the Oracle Database Documentation Library for more information on determining appropriate sizes for the SGA. This book tells how to ensure that the SGA size does not cause increased paging swapping activity. The latter is very detrimental to performance.

Once the available size of the SGA is determined, two primary tuning items need to be considered:

- Size of the shared pool
- Size of the buffer cache

An initial estimate for the shared pool size is 5 MB for each concurrent database connection previously determined.

If this estimate consumes more than 30 percent of the total SGA, use 30 percent of the total SGA instead.

Divide 60 percent of the remaining available SGA size by the block size for the database and use this value for the number of DB_BLOCK_BUFFERS. Both of these values should be initial estimates and can be refined using BSTAT/ESTAT and other RDBMS monitoring tools to determine more accurate sizes for best performance.

Other Alternatives for a Memory-Constrained System

If there is insufficient memory to run both the database and the Oracle directory server on the same computer, then one can put the database on a different computer.
Tuning Security Event Tracking

The DSA configuration attributes `orcloptracknumelemcontainers` and `orcloptrackmaxtotalsize` control how much memory is used for security event tracking.

Tuning Memory Allocated for Event Tracking

The DSA configuration set attribute `orcloptracknumelemcontainers` allows you to choose the number of in-memory cache containers to be allocated for security event tracking in the Oracle Internet Directory server. There are two subtypes for this attribute. They are `1stlevel` and `2ndlevel`. The `1stlevel` subtype is for setting the number of in-memory cache containers for storing information about users performing operations. The `2ndlevel` subtype, which is applicable only to compare operation, sets the number of in-memory cache containers for information about the users whose `userpassword` is compared and tracked when detailed compare operation statistics is programmed.

The default value of both subtypes is 255. The appropriate values for these subtypes depend on the number of users in your environment and the number of applications used to access the directory, as follows:

- In a deployment where several applications perform operations on behalf of a large number of end users, set `1stlevel` proportional to the number of applications, plus a few hundred more for end users directly accessing the directory. Then set `2ndlevel` proportional to the number of end users.
- In a deployment where end users themselves perform the operations, set `1stlevel` proportional to the number of end users, then set `2ndlevel` to a small value, such as 25.
- A typical proportional value is one fifth. Proportions between one tenth and one half are reasonable in most environments.

If your deployment requires it, set the values for `orcloptracknumelemcontainers` only when security events collection is turned on.

**Note:** After you configure the value for `1stlevel` only or values for both subtypes, security event information for one cycle of the next statistics periodicity will not be collected.

Tuning Memory Used for Each Operation

The DSA configuration attribute `orcloptrackmaxtotalsize` specifies the maximum number of bytes of RAM that security events tracking can use for each type of operation. Once the directory server exceeds this limit for information collected for an operation, the server stops collecting new information and records appropriate messages in server log files. For the compare operation, the directory server uses twice the value of the attribute, which is the combined amount of information about users performing compare operation and users whose passwords are being compared. The default value of `orcloptrackmaxtotalsize` is 100MB, which should be sufficient for most deployments. It can be increased to 200MB.

Disk Tuning

Balancing Disk I/O is an important consideration in overall RDBMS, and hence Oracle Internet Directory performance. Typically, one can maximize the I/O throughput by using one or more of the following techniques:
Database Tuning

This section describes the other tunable parameters available to an Oracle Internet Directory installation.

Table 25–2 gives a quick overview of the recommended values of RDBMS parameters for various client loads. These parameters are configurable in the initialization parameter file.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>500 Concurrent LDAP Clients</th>
<th>1000 Concurrent LDAP Clients</th>
<th>1500 Concurrent LDAP Clients</th>
<th>2000 Concurrent LDAP Clients</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open_cursors</td>
<td>200</td>
<td>200</td>
<td>200</td>
<td>200</td>
</tr>
<tr>
<td>Sessions</td>
<td>225</td>
<td>600</td>
<td>800</td>
<td>1200</td>
</tr>
<tr>
<td>Database_block_buffers</td>
<td>200 to 250 MB</td>
<td>200 to 250 MB</td>
<td>200 to 250 MB</td>
<td>200 to 250 MB</td>
</tr>
<tr>
<td>Database_block_size</td>
<td>8192</td>
<td>8192</td>
<td>8192</td>
<td>8192</td>
</tr>
<tr>
<td>Shared_pool_size</td>
<td>30 to 40 MB</td>
<td>30 to 40 MB</td>
<td>30 to 40 MB</td>
<td>30 to 40 MB</td>
</tr>
<tr>
<td>Processes</td>
<td>400</td>
<td>800</td>
<td>1000</td>
<td>1500</td>
</tr>
</tbody>
</table>

This section describes each of the RDBMS tunable parameters in more detail. It contains these topics:

- **Required Parameters**
- **Parameters Dependent on Oracle Internet Directory Server Configuration**
- **SGA Parameters Dependent on Hardware Resources**

**Required Parameters**

Configure the OPEN_CURSORS parameter as follows:

```
OPEN_CURSORS=200
```

The Oracle Database default of 50 or so is too small to accommodate Oracle Internet Directory server cursor cache. Note that this value is not dependent on other Oracle Internet Directory server parameters, such as # SERVERS and # WORKERS. The value of 200 is sufficient for any size DIT.

Oracle Internet Directory replication depends on database replication. If you are using Oracle Internet Directory replication, configure the JOB_QUEUE PROCESSES and PARALLEL_MIN_SERVERS database parameter to 1 or greater. For example:

```
JOB_QUEUE PROCESSES=1
PARALLEL_MIN_SERVERS=1
```
In Oracle Internet Directory 10g (10.1.4.0.1), the absolute minimum value you can use for `Database_block_size` is 4 KB.

**Parameters Dependent on Oracle Internet Directory Server Configuration**

Configure the SESSIOINS parameter as follows:

\[
\text{SESSIONS} = 1.1 \times \text{PROCESSES} + 5
\]

Each Oracle Internet Directory server process requires a number of concurrent database connections equal to the number of worker threads configured for that server plus one. The total number of concurrent database connections allowed must therefore include this number for each server, for each instance. The additional 20 connections added to the parameter value accounts for the Oracle background processes plus other Oracle Internet Directory processes such as OID Monitor, OID Control, Oracle directory replication server, and bulk tools.

**Using Shared Server Process**

Depending on the total number of concurrent database connections required, and as determined by the setting for the SESSIOINS parameter, enabling shared server process may help balance overall system load better. If the total number of concurrent database connections required is over 300, then configure the shared server. One shared server should be configured for every 10 database connections required.

---

**Note:**

The number of required concurrent database connections depends on the hardware selected. See *Oracle Database Net Services Administrator’s Guide* and *Oracle Database Administrator’s Guide*, both in the Oracle Database Documentation Library, for further information about the shared server configuration.

**SGA Parameters Dependent on Hardware Resources**

The main parameters that contribute to the SGA are discussed in "Memory Tuning" on page 25-5. The following are a few more parameters that may be tuned:

- **Sort area**
  
  Set to 262144 (256k) to ensure sufficient sort area available to prevent sorts on disk.

- **Redo Log Buffers**
  
  Set to 32768 (32k) as an initial estimate. If log write performance becomes a performance problem, use a large enough value to make sure (redo log space requests / redo entries) > 1/5000 to prevent the LGWR process from falling behind. This overall has little size effect on the variable SGA size, so making this a little bit too large should not be a problem.

**Entry Caching**

In Oracle Internet Directory, 10g (10.1.4.0.1), the directory server entry cache is supported only in the single directory server instance. The benefits of entry caching are maximized when the entry cache hit ratio is very high. Oracle Corporation
Optimizing Searches

This section contains these topics:

- Optimizing Searches for Large Group Entries
- Optimizing Searches for Skewed Attributes

Optimizing Subtree Searches

When searching a subtree, the server does not write to the client until a configured number of entries have been processed. This number is controlled by the attribute `orclmaxentinber` in the DSA configuration entry. By default this value is 5. If the entries are larger than 8000 bytes, reduce this value to 1.

Optimizing Searches for Large Group Entries

Searches for group entries with several thousand attribute values for either the `member` or `uniquemember` attribute can have high latency. If you find the latency unacceptably high, there are steps you can take to reduce it.

See Also: "Setting System Operational Attributes" on page 7-7 for information about attributes you set to enable and configure entry caching

Caching of Connection DNs

Each time the server performs a bind operation, it computes the list of privileged groups, including nested groups, that the identity (connection DN) performing the bind is a member of. The computed list of privilege groups for each connection DN is cached in the Oracle Internet Directory server. The number of connection DNs whose privileged groups can be cached is controlled by the attribute `orclmaxconnincache` in the DSA configuration entry. The default value is 25000 identities (connection DNs). Increase the value of `orclmaxconnincache` if your installation has more than 25000 users. For example, to increase `orclmaxconnincache` to 50000 users, you would create an LDIF file similar to this:

```ldap
dn: cn=dsaconfig,cn=configsets,cn=oracle internet directory
changetype:modify
replace: orclmaxconnincache
orclmaxconnincache: 50000
```

and then use a command line similar to this:

```
ldapmodify -p port -D cn=orcladmin -w adminPasswd -f filename
```
The simplest step is to reduce the number of attributes you are searching for. If you do not need to retrieve all the attributes of the group entry, specify required attributes in the search request to optimize the latency.

If you still see unacceptable latency, even with required attributes specified, then there are two tuning techniques you can try for reducing latency. These are known as the "Entry Cache Enabled Configuration" and "Entry Cache Disabled Configuration" respectively.

**Entry Cache Enabled Configuration**

In this technique, you try to cache the large group entry in the entry cache. To do this, increase the value of the `orclEcacheMaxEntSize` attribute in the Oracle Internet Directory server configuration entry (cn=dsaconfig,cn=configsets,cn=oracle internet directory). This attribute controls the maximum size of a cache entry. The default value is 1M. If the size of the large group entry is greater than the value of `orclEcacheMaxEntSize`, change it to a large enough value to ensure that the large group entry is cached.

---

**Note:**
- If you expect frequent updates to large groups, then do not use this tuning methodology. Use “Entry Cache Disabled Configuration” instead.
- If you are using a multiserver LDAP instance or if this server is part of a high availability configuration with Oracle Internet Directory servers on different machines connecting to the same database, then the entry cache is automatically disabled Use "Entry Cache Disabled Configuration" instead.

---

**Entry Cache Disabled Configuration**

In this technique, you disable the entry cache by setting the value of the attribute `orclEcacheenabled` to 0 (zero) in the root DSE entry. The steps are as follows:

1. Stop the Oracle Internet Directory instance.
2. Log in to the Oracle Internet Directory database as the ODS user and execute the following query:

   ```sql
   DROP INDEX EI_ATTRSTORE;
   CREATE BITMAP INDEX EI_ATTRSTORE ON DS_ATTRSTORE(ENTRYID,ATTRNAME) TABLESPACE OLTS_ATTRSTORE PARALLEL COMPUTE STATISTICS NOLOGGING;
   ```
3. Start the Oracle Internet Directory instance.

---

**Note:** Employ this method carefully and only when required, as re-creating `EI_ATTRSTORE` in this fashion may degrade the performance of typical messaging search operations by 5-10%.

---

**Optimizing Searches for Skewed Attributes**

To service a typical search request, the directory server sends a SQL statement to the Oracle Database. If a given attribute has very different response times depending on its value, then the attribute is said to be skewed. For example, if searches for `my_`...
attribute=value1 and my_attribute=value2 have very different response
times, then my_attribute is said to be a skewed.

You can uniform the response times for searches for such an attribute by adding it as a
value of the orclskewedattribute attribute, which is in the dsaconfig entry. The
DN of the dsaconfig entry is cn=dsaconfig,cn=configsets,cn=oracle
internet directory.

By default, the objectclass attribute is listed as a value in the
orclskewedattribute attribute.

Optimizing Searches for Skewed Attributes by Using Oracle Directory Manager

To optimize queries to the database:

1. In the navigator pane, expand Oracle Internet Directory Servers and select the
directory server instance.

2. In the right pane, select the Query Optimization tab. The fields in the Query
Optimization tab page are listed and described in Table A-44 on page A-29.

3. In the Query Optimization tab page, in the Attributes with Low Cardinality field, enter the attributes you want to designate as skewed.

4. Choose Apply.

Optimizing Searches for Skewed Attributes by Using ldapmodify

To optimize the search for a skewed attribute, you use ldapmodify to add it as a value
of the orclskewedattribute attribute.

For example, to add my_attribute to the orclskewedattribute attribute, you
would enter the following:

```bash
ldapmodify -D "cn=orcladmin" -w password -h host -p port <<!
dn: cn=dsaconfig,cn=configsets,cn=oracle internet directory
changetype: modify
add: orclskewedattribute
orclskewedattribute: my_attribute
!
```

Setting the Time Limit Mode

When you set the server operation time limit as described in "Setting System
Operational Attributes" on page 7-7, you specified the maximum number of seconds
allowed for a search to be completed. To adjust server performance, you can also set
the search time limit mode to be either accurate or approximate. If you specify it as
accurate, then searches end precisely at the specified number of seconds. If you specify
it as approximate, then searches end within a few seconds of the specified number of
seconds. In smaller workloads, the latter provides better performance.

Setting the Time Limit Mode by Using Oracle Directory Manager

To set the time limit mode:

1. In the navigator pane, expand Oracle Internet Directory Servers and select the
directory server instance.

2. In the right pane, select the Query Optimization tab.

3. In the Query Optimization tab page, in the Time Limit Mode field, select either
Accurate or Approximate.
4. Choose Apply.

Setting the Time Limit Mode by Using ldapmodify

To specify the search time limit mode to be either accurate or approximate, you set the orcltlimitmode attribute. A value of 0 is accurate, and a value of 1 is approximate. The default value is 1.

Setting the Timeout for Client/Server Connections

You can specify the amount of idle time allowed for connections between clients and the directory server. To set the timeout for client/server connections:

1. In the navigator pane, expand Oracle Internet Directory Servers and select the directory server instance.
2. In the right pane, select the Query Optimization tab.
3. In the Query Optimization tab page, in the LDAP Connection Timeout field, enter the maximum number of seconds that the directory client can remain idle before the connection is terminated. The default is 0, meaning that there is no timeout.
4. Choose Apply.

Setting the Timeout for Write Operations

When an LDAP client initiates an operation, then does not respond to the server for a configured number of seconds, the server closes the connection. The number of seconds is controled by the attribute orclnwrwtimeout in the DSA configuration entry. The default is 300 seconds. To increase it to 500 seconds, you would create an LDIF file similar to this:

dn: cn=dsaconfig,cn=configsets,cn=oracle internet directory
changetype:modify
replace: orclnwrwtimeout
orclnwrwtimeout: 500

and then use a command line similar to this:

ldapmodify -p port -D cn=orcladmin -w adminPasswd -f filename
The term "garbage" refers to any data not needed by the directory but still occupying space on it. This unwanted or obsolete data can eventually fill up the disk and decrease directory performance. The process of removing this unwanted data from the directory is called garbage collection.

This chapter contains these topics:

- About the Oracle Internet Directory Garbage Collection Framework
- Modifying Oracle Internet Directory Garbage Collectors
- Enabling, Disabling, and Monitoring Logging for Oracle Internet Directory Garbage Collectors

About the Oracle Internet Directory Garbage Collection Framework

A garbage collector is a background database process that removes unwanted data from the directory. The Oracle Internet Directory garbage collection framework provides a default set of garbage collectors, and enables you to modify them. The Oracle Internet Directory statistics collector also uses the Oracle Internet Directory garbage collection framework.

This section contains these topics:

- Components of the Oracle Internet Directory Garbage Collection Framework
- How Oracle Internet Directory Garbage Collection Works
- Garbage Collector Entries and the Oracle Internet Directory Statistics Collector Entry
- Change Log Purging in Multimaster Replication

Components of the Oracle Internet Directory Garbage Collection Framework

This section describes the components that make up the Oracle Internet Directory garbage collection framework, namely, the garbage collection plug-in and the background database processes.

Garbage Collection Plug-in

Garbage collection in Oracle Internet Directory relies on a garbage collection plug-in that receives requests to manage garbage collectors. This plug-in is installed with
Oracle Internet Directory, and is enabled by default. The entry for this plug-in is cn=plugin,cn=subconfigsubentry.

This plug-in has three triggers:

- The DN of the plug-in trigger used to create a garbage collection job is: cn=Add PurgeConfig,cn=plugin,cn=subconfigsubentry.
- The DN of the plug-in trigger used to modify a garbage collection job is: cn=Modify PurgeConfig,cn=plugin,cn=subconfigsubentry.
- The DN of the plug-in trigger used to delete a garbage collection job is: cn=Delete PurgeConfig,cn=plugin,cn=subconfigsubentry.

See Also: "Oracle Internet Directory Configuration Schema Elements" in Oracle Identity Management User Reference for a list and descriptions of the attributes of the garbage collection plug-in

Background Database Processes
The background database processes that are invoked by the garbage collection plug-in include garbage collectors and the Oracle Internet Directory statistics collector.

Garbage Collectors You can set and manage these behaviors of a garbage collector:

- The time it starts
- The age of the data you want it to purge
- How often it runs
- The type of data you want it to purge
- The number of entries to purge at a time

Predefined Garbage Collectors A default installation of Oracle Internet Directory includes these predefined garbage collectors:

- Audit log garbage collector: Cleans up obsolete entries created for auditing the directory. The container for this garbage collector is cn=auditlog purgeconfig,cn=purgeconfig,cn=subconfigsubentry.
- Change log garbage collector: Cleans up the consumed change log entries in the directory. The container for this garbage collector is cn=changelog purgeconfig, cn=purgeconfig,cn=subconfigsubentry.
- General statistics garbage collector: Cleans up obsolete entries created by Oracle Internet Directory Server Manageability for monitoring general statistics of the directory. The container for this garbage collector is cn=general stats purgeconfig, cn=purgeconfig,cn=subconfigsubentry.
- Health statistics garbage collector: Cleans up obsolete entries created by Oracle Internet Directory Server Manageability for monitoring health statistics of the directory. The container for this garbage collector is cn=health stats purgeconfig, cn=purgeconfig,cn=subconfigsubentry.
- Security and refresh events garbage collector: Cleans up obsolete entries created by Oracle Internet Directory Server Manageability for monitoring security and refresh events of the directory. The container for this garbage collector is
About the Oracle Internet Directory Garbage Collection Framework

- **System resource events garbage collector**: Cleans up obsolete entries created by Oracle Internet Directory Server Manageability for monitoring system resource events of the directory. The container for this garbage collector is `cn=sysresource events purgeconfig, cn=purgeconfig, cn=subconfigsubentry`.

- **Tombstone garbage collector**: Cleans up obsolete entries marked as deleted in the directory. The container for this garbage collector is `cn=tombstone purgeconfig, cn=purgeconfig, cn=subconfigsubentry`.

- **LDAP performance monitoring garbage collector**: Cleans up LDAP server performance statistics data. The container for this garbage collector is `cn=perfstats purgeconfig, cn=purgeconfig, cn=subconfigsubentry`.

- **LDAP bind performance monitoring garbage collector**: Cleans up bind performance data gathered for security events tracking. The container for this garbage collector is `cn=bindsec stats purgeconfig, cn=purgeconfig, cn=subconfigsubentry`.

- **LDAP compare performance monitoring garbage collector**: Cleans up compare performance data gathered for security events tracking. The container for this garbage collector is `cn=comparesec stats purgeconfig, cn=purgeconfig, cn=subconfigsubentry`.

- **LDAP compare failure performance monitoring garbage collector**: Cleans up compare failure performance data gathered for security events tracking. The container for this garbage collector is `cn=comparefailure stats purgeconfig, cn=purgeconfig, cn=subconfigsubentry`.

**Oracle Internet Directory Statistics Collector** You can set and manage these behaviors of the Oracle Internet Directory statistics collector:

- The time it starts
- How often it runs

The Oracle Internet Directory statistics collector collects statistics about Oracle Internet Directory. The container for this background database process is `cn=oidstats_config, cn=purgeconfig, cn=subconfigsubentry`.

**Note:** Oracle recommends that you not delete any of the predefined garbage collectors. Deleting one or more of them can result in the proliferation of obsolete data, eventually exhausting all the available disk space.

You may, however, modify predefined garbage collectors to customize their behavior.

**See Also:**
- "Capabilities of Oracle Internet Directory Server Manageability" on page 14-14
- "Oracle Internet Directory Configuration Schema Elements" in *Oracle Identity Management User Reference*
How Oracle Internet Directory Garbage Collection Works

Figure 26–1 shows an example of a garbage collector operation that purges change log entries.

Figure 26–1 Example: Garbage Collection of Change Log Entries

As the example in Figure 26–1 on page 26-4 shows, the garbage collection process is as follows:

1. An LDAP client sends to the directory server a request for a particular garbage collection operation. The operation could be, for example, to purge the entries of tombstones, change logs, or audit logs.

2. The directory server passes the request to the garbage collection plug-in.

3. The garbage collection plug-in sends the request to the garbage collection engine in the Oracle Internet Directory-designated database.

4. The garbage collection engine triggers the corresponding background database process—in this case, the change log garbage collector. The background database process runs according to the parameters specified in its configuration set entry.
Garbage Collector Entries and the Oracle Internet Directory Statistics Collector Entry

Garbage collector entries, each with attributes specifying how it is to behave, are located in the entry `cn=purgeconfig`, which is located immediately below the entry `cn=subconfigsubentry`.

See Also: "Oracle Internet Directory Configuration Schema Elements" in Oracle Identity Management User Reference for a description of each garbage collector attribute.

The Oracle Internet Directory statistics collector entry, with its attributes, is also located in the entry `cn=purgeconfig`, immediately below the entry `cn=subconfigsubentry`.

Figure 26–2 shows the location of these entries.

Change Log Purging in Multimaster Replication

Both replication and Oracle Directory Integration Platform use change logs to propagate information from a supplier directory to a consumer directory. All change logs are stored in the table `ods_chg_log`. In addition, replication change logs are stored in `asr_chg_log`. When the change log garbage collector runs, it purges change logs that are no longer needed by any change log consumers. This prevents the change log store in the Oracle Internet Directory database from becoming too large.

Change Log Purging Methods

The change log garbage collector uses the following two methods to determine which change logs to purge:

- Change number-based purging
  
  Change number-based purging respects the change status of all change log consumers. That is, it does not purge change logs unless they have been consumed by all consumers. When the change log garbage collector runs, it purges all change logs that have been consumed by replication, Oracle Directory Integration Platform, and other consumers.

- Time-based purging
Time-based purging is a fall-back method designed to purge change logs of a certain age. It ensures that old change logs are purged even if they have not been consumed by all change log subscribers. Time-based purging respects the change status of replication, but not the change status of other consumers. The change log garbage collector purges all change logs that are not needed by replication and that are at least orclpurgetargetage hours old. If orclpurgetargetage is zero, the change log garbage collector does this immediately. If orclpurgetargetage is an invalid number or not defined, the default value is 240 hours (10 days). Change logs needed by replication are not purged until they have been consumed by replication.

If you have deployed Oracle Directory Integration Platform, and you want to enable time-based purging, be sure to set orclpurgetargetage to a large enough value to allow change logs to be processed by Oracle Directory Integration Platform before they are purged. A value of 240 will allow 10 days before change logs are purged.

**Configuring Change Log Purging**

You configure time-based purging by modifying the orclpurgetargetage attribute of the changelog purging configuration entry. This example configures time-based purging for 120 hours (5 days). Use an LDIF file similar to this:

```ldiff
dn: cn=changelog purgeconfig,cn=purgeconfig,cn=subconfigsubentry
changetype:modify
replace: orclpurgetargetage
orclpurgetargetage: 120
```

To apply the ldif file `mod.ldif`, type:

```
ldapmodify -p port -h host -D dn -w password -f mod.ldif
```

---

**Note**: The container for the change log garbage collector is `cn=changelog purgeconfig, cn=purgeconfig, cn=subconfigsubentry`.

**See Also**: "Oracle Internet Directory Configuration Schema Elements" in Oracle Identity Management User Reference

---

## Modifying Oracle Internet Directory Garbage Collectors

This section contains these topics:

- Modifying a Garbage Collector by Using Oracle Directory Manager
- Modifying a Garbage Collector by Using Command-Line Tools

### Modifying a Garbage Collector by Using Oracle Directory Manager

To modify a garbage collector:

1. In the navigator pane, expand in succession Oracle Internet Directory Servers, directory server instance, Garbage Collection Management, then select the garbage collector you want to configure. The Garbage Collector Window appears in the right pane.
2. In the Garbage Collector window, enter the values for this garbage collector. These fields are described in Table A–9 on page A-6.

3. Choose Apply.

Modifying a Garbage Collector by Using Command-Line Tools

This section provides examples of how to modify garbage collectors by using command-line tools. The garbage collection attributes that you can modify are listed in "Oracle Internet Directory Configuration Schema Elements" in Oracle Identity Management User Reference.

Example 1: Modifying a Garbage Collector
Suppose that you want the tombstone garbage collector to run immediately. The LDIF would look like this:

dn: cn=tombstone purgeconfig, cn=purge config, cn=subconfigsubentry
changetype:modify
replace: orclpurgenow
orclpurgenow: 1

Load this entry with ldapmodify.

ldapmodify -h hostname -p port# -D username -w passwd \
-f file_name_of_defined_entry

Example 2: Disabling a Garbage Collector Change Log
Suppose that you want to disable changelog garbage collector.

dn: cn=changelog purgeconfig, cn=purgeconfig, cn=subconfigsubentry
changetype: modify
replace: orclpurgeenable
orclpurgeenable: 0

Load this entry with ldapmodify.

ldapmodify -h hostname -p port# -D username -w passwd \
-f file_name_of_defined_entry

Modifying the Oracle Internet Directory Statistics Collector

You modify the Oracle Internet Directory statistics collector in the same way as the garbage collectors, but there are only three modifiable fields. See "Oracle Internet Directory Statistics Collection Management Fields in Oracle Directory Manager" on page A-6.

Enabling, Disabling, and Monitoring Logging for Oracle Internet Directory Garbage Collectors

This section contains these topics:

- Enabling Logging for Oracle Internet Directory Garbage Collectors
- Disabling Logging for Oracle Internet Directory Garbage Collectors
- Monitoring Garbage Collection Logging
Enabling Logging for Oracle Internet Directory Garbage Collectors

If you enable logging for garbage collectors, then the directory server writes the information into a file in the file system. This information includes:

- The job identifier
- A job description of the garbage collector
- The number of entries purged
- The operation status
- The time stamp
- Any errors caught

To enable logging of garbage collection information:

1. Set the orclpurgedebug attribute to 1 if needed. When orclpurgedebug is set to 1, extra debugging detail information will be logged. This is useful for troubleshooting garbage collection problems.
2. Set the orclpurgefilename attribute to a valid file name for the log file, for example: oidgc001.log.
3. Set the orclpurgefileloc attribute to the path name of the directory in which the log file is located, for example: /private/qzhou/oracle/ldap/log.
4. Enable PL/SQL I/O access to the directory specified in step 3. To do this, include the following in the database:

   \[ UTL_FILE_DIR=PATH_NAME \]

   where PATH_NAME is the path you specified in Step 3.

   **See Also:** The section on the UTL_FILE_DIR parameter type in the Oracle Database Reference

5. Shut down the replication server, then the Oracle Internet Directory server.
6. Restart the database.
7. Start the Oracle Internet Directory server, then the replication server.

Disabling Logging for Oracle Internet Directory Garbage Collectors

To disable logging of garbage collection information, set the orclpurgedebug attribute to 0.

**Note:** Even when orclpurgedebug is set to 0, minimal information about garbage collector operation is still logged to indicate the garbage collector's activities.

Monitoring Garbage Collection Logging

The information in the garbage collection log can be useful for monitoring and troubleshooting garbage collection. You determine the location of the log by setting attributes when enabling logging. For example, if you configured:

orclpurgefilename = oidgc001.log
orclpurgefileloc = /private/qzhou/oracle/ldap/log
when you enabled logging, then you can monitor change log garbage collection activities by reading the file
/private/qzhou/oracle/ldap/log/oidgc001.log.

The following is an example of the information logged when an administrator modified the orclpurgenow attribute of the change log garbage collection configuration entry:

Running Garbage Collector: cn=changelog purgeconfig
Starting time: 2005/03/24 11:03:23
PurgeConfig object located, Eid= 936
purge_ODSChglog: Nothing to be purged(no_work_to_do)
purge_ODSChglog: 107 chglogs successfully purged
purge_ASRChglog: Nothing to be purged(no_work_to_do)
purge_ASRChglog: 0 chglogs successfully purged
purge_ASRChglog: 0 chglogs successfully purged

Modifying Garbage Collector for at "2005/03/24 11:03:23
Garbage Collector DN recognized, rdn=cn=changelog purgeconfig
orclPurgeNow successfully retrieved.
Garbage Collector job found: jobno=21
Garbage Collector has been run
Garbage collector is updated successfully!

Modifying orclpurgenow forces the change log garbage collector to run immediately. As shown in the first paragraph, 107 change logs were purged from the ods_chg_log table and 0 change logs were purged from the asr_chg_log table. Also, the information in the second paragraph indicates successful modification of orclpurgenow attribute.
This chapter explains how to migrate data from both LDAP Version 3-compatible directories and application-specific data repositories into Oracle Internet Directory.

This chapter contains these topics:

- The Default Directory Structure of Oracle Internet Directory
- Migrating Data from LDAP-Compliant Directories
- Migrating User Data from Application-Specific Repositories

The Default Directory Structure of Oracle Internet Directory

During an Oracle Internet Directory installation, Oracle Universal Installer creates a default schema and directory information tree (DIT). This default DIT framework is described in Chapter 3, "Directory Concepts and Architecture" and in Chapter 23, "Deployment of Oracle Identity Management Realms". The framework is flexible and you can modify it to suit the needs of your deployment.

In Oracle Internet Directory 10g (10.1.4.0.1), the following directory elements are created by default:

- Root Oracle Context (cn=OracleContext): This is the container where Oracle products store enterprise-wide configuration data.

- Default identity management realm (dc=dns_domain_of_host, dc=com): This is the container under which Oracle products expect to find enterprise users and groups. It approximates the enterprise DIT structure. For example, if Oracle Internet Directory is installed on a computer whose host name is: my_computer.us.my_company.com, then the default identity management realm created at installation of Oracle Internet Directory would be dc=us, dc=my_company, dc=com. Oracle products expect to find all users under the container cn=users, dc=us, dc=my_company, dc=com and all groups under cn=groups, dc=us, dc=my_company, dc=com. In addition to creating the default identity management realm entry, the Oracle Internet Directory Configuration Assistant stores a pointer to it in the Root Oracle Context so that other Oracle Internet Directory-enabled components can bootstrap themselves.

You can change this default identity management realm to suit your deployment requirements.
Migrating Data from LDAP-Compliant Directories

This section provides practical information for migrating data from an LDAP-compliant, third-party directory to Oracle Internet Directory. If you have a directory with an already-established structure, and you want to migrate the data from that directory into the default directory structure environment, then follow the instructions in this section.

This section contains these topics:

- Tools
- Common Usage Scenarios
- Tasks For Migrating Data from LDAP-Compliant Directories

Tools

Three tools are commonly used for migrating third-party LDAP data to Oracle Internet Directory. They are:

- bulkload
- dipassistant
- Oracle Directory Integration Platform Server

Which tool or tools you use depends on several factors, including the size of the data you are importing and whether the data must be mapped.

bulkload

The Bulk Loading Tool, bulkload, is a command-line tool for loading a large number of entries into a directory server. It uses Oracle SQL*Loader to load the directory entries. The bulkload tool expects the input file to be in LDAP Data Interchange Format (LDIF). The bulkload tool can validate LDIF input for referential integrity, but it cannot perform any mapping or other transformation on the data.

Table 27–1, "Features of bulkload and dipassistant" lists the features of bulkload, as compared with dipassistant bootstrap. For bulkload syntax information and examples, see "Oracle Internet Directory Data Management Tools" in Oracle Identity Management User Reference.

dipassistant

The Directory Integration Assistant, dipassistant, is a command-line tool for administering the synchronization profiles scheduled by the Oracle directory integration server. An administrator can use the dipassistant bootstrap operation to perform the initial migration of data between a connected directory and Oracle Internet Directory when configuring the Oracle directory integration server to perform ongoing synchronization. You also use it for a one-time data migration, without ongoing synchronization.

The dipassistant bootstrap operation can take data either directly from a third-party LDAP-compliant directory or from an LDIF file, tagged file, or CSV file. You must provide mapping rules, either as a synchronization profile or in a configuration file.

Table 27–1, "Features of bulkload and dipassistant" lists the features of dipassistant bootstrap, as compared with bulkload. For dipassistant syntax information, configuration file properties, information about input file types, and examples, see "Oracle Directory Integration Platform
Tools” in Oracle Identity Management User Reference and Oracle Identity Management Integration Guide.

Table 27–1 Features of bulkload and dipassistant

<table>
<thead>
<tr>
<th>Feature</th>
<th>bulkload</th>
<th>dipassistant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Speed</td>
<td>Fast</td>
<td>Slow</td>
</tr>
<tr>
<td>Data transfer method</td>
<td>SQL</td>
<td>LDAP</td>
</tr>
<tr>
<td>Input types accepted</td>
<td>LDIF file only</td>
<td>LDIF file, LDAP directory, tagged file, CSV file</td>
</tr>
<tr>
<td>Transforms data</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Validates LDIF input</td>
<td>Yes</td>
<td>No</td>
</tr>
</tbody>
</table>

Oracle Directory Integration Platform Server

Under some circumstances, an administrator might choose not to use dipassistant when configuring the Oracle directory integration server. Once configured, the Oracle directory integration server itself can migrate data from a connected directory to Oracle Internet Directory. You can also use the Oracle directory integration server for a one-time data migration. For more information, see Oracle Identity Management Integration Guide.

Common Usage Scenarios

This section describes different scenarios for using the tools described in the "Tools" section. They include:

- Scenario 1: Using an LDIF File and bulkload
- Scenario 2: Using dipassistant Directly
- Scenario 3: Using an LDIF File and dipassistant
- Scenario 4: Using dipassistant, bulkload, and LDIF Files
- Scenario 5: Using the Oracle Directory Integration Platform Server

Scenario 1: Using an LDIF File and bulkload

When no translation is required and data is very large (500,000 or more), bulkload is the best choice for migrating data from a third-party directory to Oracle Internet Directory. It is fast and it can validate LDIF input. To use this method, you must first export data from the third-party directory to an LDIF file, as shown in Figure 27–1, "Using an LDIF File and bulkload".

Figure 27–1 Using an LDIF File and bulkload

LDIF is the IETF-sanctioned ASCII interchange format for representing LDAP-compliant directory data as a file. All LDAP-compliant directories should have
tools to export their contents into one or more LDIF files representing the DIT at the time of export.

See Also: RFC 2849 of the IETF, available for download at:
http://www.ietf.org

When using an LDIF file and bulkload to migrate data to Oracle Internet Directory, you must perform the following tasks.

- Task 1: Export Data from the Non-Oracle Internet Directory Server into LDIF File Format
- Task 2: Analyze the LDIF User Data for Any Required Schema Additions Referenced in the LDIF Data
- Task 3: Extend the Schema in Oracle Internet Directory
- Task 4: Remove Any Proprietary Directory Data from the LDIF File
- Task 5: Remove Operational Attributes from the LDIF File
- Task 6: Remove Incompatible userPassword Attribute Values from the LDIF File
- Task 7: Run the bulkload check="TRUE" Mode and Determine Any Remaining Schema Violations or Duplication Errors

These tasks are explained in “Tasks For Migrating Data from LDAP-Compliant Directories” on page 27-6.

Scenario 2: Using dipassistant Directly

If you must perform mapping when migrating the data from the third-party directory to Oracle Internet Directory, and if the data is small in size, you can use dipassistant. As shown in Figure 27–2, "Using dipassistant Directly", you can use the third-party directory itself as input to dipassistant.

![Figure 27–2 Using dipassistant Directly](image)

Scenario 3: Using an LDIF File and dipassistant

Scenario 3 is a variation on Scenario 2. If you do not have direct access to the third-party directory, you can have the administrator export the data to an LDIF file. As shown in Figure 27–3, "Using an LDIF File and dipassistant", dipassistant can take its input from an LDIF file. You could also use Oracle directory integration server to migrate the data.
Whenever you use an LDIF file and bulkload to migrate data to Oracle Internet Directory, you must perform certain tasks. In this scenario, you will be using a mapping file with dipassistant or Oracle Directory Integration Platform, so do not have to perform all the tasks listed in "Scenario 1: Using an LDIF File and bulkload". You only have to perform the following tasks:

- Task 1: Export Data from the Non-Oracle Internet Directory Server into LDIF File Format
- Task 2: Analyze the LDIF User Data for Any Required Schema Additions Referenced in the LDIF Data
- Task 3: Extend the Schema in Oracle Internet Directory

Scenario 4: Using dipassistant, bulkload, and LDIF Files

If you have a large amount of data and you must perform mapping on the data, you can use a combination of tools. As shown in Figure 27–4, "Using dipassistant, bulkload, and LDIF Files", you can export the data from the third-party directory to an LDIF file, then use dipassistant bootstrap to perform the mapping into another LDIF file, which you then load with bulkload.

As in Scenario 3: Using an LDIF File and dipassistant, you only have to perform these tasks:

- Task 1: Export Data from the Non-Oracle Internet Directory Server into LDIF File Format
- Task 2: Analyze the LDIF User Data for Any Required Schema Additions Referenced in the LDIF Data
- Task 3: Extend the Schema in Oracle Internet Directory
Scenario 5: Using the Oracle Directory Integration Platform Server

The Oracle directory integration server allows you to configure bidirectional, ongoing integration between Oracle Internet Directory and a Third-party directory, as shown in Figure 27–5, “Using the Oracle Directory Integration Server”. For more information, see Oracle Identity Management Integration Guide.

**Figure 27–5 Using the Oracle Directory Integration Server**

Tasks For Migrating Data from LDAP-Compliant Directories

To migrate data from LDAP-compliant directories using an LDIF file, you perform the tasks explained in this section.

As described in "Common Usage Scenarios" on page 27-3, you do not need to perform Tasks 4-7 if you’re using dipassistant or Oracle Directory Integration Platform with a mapping file.

| Task 1: Export Data from the Non-Oracle Internet Directory Server into LDIF File Format |
| Task 2: Analyze the LDIF User Data for Any Required Schema Additions Referenced in the LDIF Data |
| Task 3: Extend the Schema in Oracle Internet Directory |
| Task 4: Remove Any Proprietary Directory Data from the LDIF File |
| Task 5: Remove Operational Attributes from the LDIF File |
| Task 6: Remove Incompatible userPassword Attribute Values from the LDIF File |
| Task 7: Run the bulkload check="TRUE" Mode and Determine Any Remaining Schema Violations or Duplication Errors |

Task 1: Export Data from the Non-Oracle Internet Directory Server into LDIF File Format

See the vendor-supplied documentation for instructions. If flags or options exist for exporting data from the foreign directory, be sure to select the method that:

- Produces LDIF output with the least amount of proprietary information included
- Provides maximum conformance to the IETF Request for Comments 2849 of the IETF, available for download at: [http://www.ietf.org](http://www.ietf.org)

Task 2: Analyze the LDIF User Data for Any Required Schema Additions Referenced in the LDIF Data

Any attributes not found in the Oracle Internet Directory base schema require extension of the Oracle Internet Directory base schema prior to the importation of the LDIF file. Some directories may support the use of configuration files for defining extensions to their base schema (Oracle Internet Directory does not). If you have a
configuration file you can use it as a guideline for extending the base schema in Oracle Internet Directory in "Task 3: Extend the Schema in Oracle Internet Directory".

**Task 3: Extend the Schema in Oracle Internet Directory**

See Chapter 11, "Directory Schema Administration" for tips on how to extend the directory schema in Oracle Internet Directory. You can do this by using either Oracle Directory Manager or the SchemaSynch tool, which is documented in Oracle Identity Management User Reference.

If you have users who will be using other Oracle products, you must create users with object classes `orclUserV2` and its required attributes. If you are integrating with Active Directory, you must create users with object class `orclADUser` and its required attributes. These object classes and their attributes are documented in Oracle Identity Management User Reference.

**Task 4: Remove Any Proprietary Directory Data from the LDIF File**

Certain elements of the LDAP v3 standard have not yet been formalized, such as ACI attributes. As a result, various directory vendors implement ACI policy objects in ways that do not translate well across vendor installations.

After the basic entry data has been imported from the cleaned up LDIF file to Oracle Internet Directory, you must explicitly reapply security policies in the Oracle Internet Directory environment. You can do this by using either Oracle Directory Manager, or command-line tools and LDIF files containing the desired ACP information.

There may be other proprietary metadata unrelated to access control. You should remove this as well. Understanding the various IETF RFCs can help you determine which directory metadata is proprietary to a given vendor and which complies with the LDAP standards, and is thus portable by way of an LDIF file.

**Task 5: Remove Operational Attributes from the LDIF File**

Four of the standard LDAP v3 operational attributes, namely, creatorsName, createTimestamp, modifiersName, and modifyTimestamp are automatically generated by Oracle Internet Directory whenever entries are created or imported. It is not possible to instantiate these values from existing directory data, for example by using LDIF file importation. Therefore you should remove these attributes from the file before attempting to import.

**Task 6: Remove Incompatible userPassword Attribute Values from the LDIF File**

Oracle Internet Directory 10g (10.1.4.0.1) supports the following userPassword attribute hash algorithms:

- No encryption
- MD4
- MD5
- SHA
- SSHA
- UNIX Crypt

The userPassword attribute hash values used by some vendor products are not compatible with Oracle Internet Directory. As a result, you must remove all lines corresponding to the userPassword attribute and value from the LDIF data file unless they are represented in plain text or contain no value. After importation of the
LDIF data, you must manually re-enter or upload hashed userPassword information separately into the directory. Be sure that the passwords comply with the Oracle Internet Directory password policies and are in clear text.

**Task 7: Run the `bulkload check="TRUE"` Mode and Determine Any Remaining Schema Violations or Duplication Errors**

Before generating and loading an LDIF file, always perform a check on it by using the `bulkload` utility check mode. The `bulkload` output reports any inconsistencies in the data.

**See Also:** The `bulkload` command-line tool reference in *Oracle Identity Management User Reference* for instructions on how to use the `bulkload` check mode

---

**Migrating User Data from Application-Specific Repositories**

Migrating user data from an application-specific repository requires:

- Collecting the user data from the application-specific repository and formatting it in a way that the directory can read it

- Making that data available to the directory administrator who must then:
  - Specify where to place it in the directory
  - Import it into the directory

---

**The Intermediate Template File**

To enable this migration to happen, the Oracle Directory Provisioning Integration Service requires the application-specific repository to export its data to an intermediate template file. Records in this template file are not in pure LDIF; they contain substitution variables that have to do with, for example, the location in the directory where the information is finally to reside. The application leaves these variables undefined, so that you, the directory administrator can define them later on.

To convert the user data from this intermediate template file into proper LDIF, you use the OID Migration Tool (`ldifmigrator`). Once the data is converted to LDIF, you can load it into the directory.

To summarize: Migrating data from application-specific repositories involves these general steps:

1. Exporting the application-specific data as an intermediate template file

2. You, the directory administrator, using the OID Migration Tool (`ldifmigrator`) to read these partial LDIF entries and convert them to pure LDIF entries based on the deployment choices

3. You, the directory administrator, loading the data, now in pure LDIF, into Oracle Internet Directory

4. The application completing the migration process according to its own specifications
Reconciling Data in Application Repository with Data Already in Oracle Internet Directory

The data you are migrating from an application-specific repository may already reside in Oracle Internet Directory. If this is the case, then you can reconcile differences between the two directories by using the reconciliation feature of the OID Migration Tool (ldifmigrator).

See Also: The ldifmigrator command-line tool reference in Oracle Identity Management User Reference for information about the reconciliation feature of the OID Migration Tool.

Tasks For Migrating Data from Application-Specific Repositories

To migrate data from application-specific repositories, you create an intermediate template file, then run the OID Migration Tool.

Task 1: Create an Intermediate Template File


When generating the intermediate template file, migrating applications must list all user records sequentially with a record separator as defined in RFC 2849. The OID Migration Tool (ldifmigrator) assigns all of these users to the default identity management realm, which corresponds to the enterprise itself.

Figure 27–6 shows the overall structure of the intermediate template file containing user entries.

Figure 27–6  Structure of the Intermediate User File

```
User Entry 1
User Entry 2
User Entry 3
User Entry N
```

The intermediate template file uses the following format to generate a valid user entry. All of the strings in bold text are supplied from the application-specific repository.

dn: cn=UserID, %s_UserContainerDN%
sn: Last_Name
orclGlobalID: GUID_for_User
%s_UserNicknameAttribute%: UserID
objectClass: inetOrgPerson
objectClass: orclUserV2

In this template, the strings %s_UserContainerDN% and %s_UserNicknameAttribute% are substitution variables for which the OID Migration Tool provides values. The OID Migration Tool determines these values according to
deployment-specific considerations. Either the application passes the arguments to the OID Migration Tool, or the tool retrieves them from the directory.

**Example: User Entries in an Intermediate Template File** The following intermediate template file includes user entries generated by the application-specific migration logic. In this example, all of the data listed in **bold text** is from the application-specific user repository.

```plaintext
dn: cn=jdoe, %s_UserContainerDN%
sn: Doe
%s_UserNicknameAttribute%: jdoe
objectClass: inetOrgPerson
objectClass: orclUserV2
title: Member of Technical Staff
homePhone: 415-584-5670
homePostalAddress: 234 Lez Drive$ Redwood City$ CA$ 94402

---

dn: cn=jsmith, %s_UserContainerDN%
sn: Smith
%s_UserNicknameAttribute%: jsmith
objectClass: inetOrgPerson
objectClass: orclUserV2
title: Member of Technical Staff
homePhone: 650-584-5670
homePostalAddress: 232 Gonzalez Drive$ San Francisco$ CA$ 94404

---

dn: cn=lrider, %s_UserContainerDN%
sn: Rider
%s_UserNicknameAttribute%: lrider
objectClass: inetOrgPerson
objectClass: orclUserV2
title: Senior Member of Technical Staff
homePhone: 650-584-5670
```

Once all of the user data is converted to the intermediate file format, the OID Migration Tool further converts it into a proper LDIF file that can be loaded into Oracle Internet Directory.

You can find examples of intermediate template files in `$ORACLE_HOME/ldap/schema/oid`.

**Attributes in User Entries** Each user entry has mandatory and optional attributes.

Table 27–2 lists and describes the mandatory attributes in a user entry.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dn</td>
<td>Distinguished name of the user entry with appropriate substitution variables. The relative distinguished name of the entry MUST contain the <code>cn</code> attribute.</td>
</tr>
<tr>
<td>sn</td>
<td>Surname—that is, the last name—of the user</td>
</tr>
<tr>
<td>objectclass</td>
<td>Object classes the entry should minimally belong to: <code>inetOrgPerson</code> and <code>orclUserV2</code></td>
</tr>
</tbody>
</table>
Task 2: Run the OID Migration Tool
Once you have set up the intermediate template file, the OID Migration Tool enables you to bring all pertinent data from the application-specific repository into Oracle Internet Directory. Once you have migrated the data, you can update whatever portion of it is relevant to the application by synchronizing that application with Oracle Internet Directory. You synchronize by using the Oracle Directory Synchronization Service.

See Also: The ldifmigrator command-line tool reference in Oracle Identity Management User Reference for instructions about using the OID Migration Tool
Directory server chaining is a new feature of Oracle Internet Directory, introduced at 10g (10.1.4.0.1). It was implemented using the new Java Plug-in framework.

Server chaining enables you to map entries that reside in third party LDAP directories to part of the directory tree and access them through Oracle Internet Directory, without synchronization or data migration. With server chaining, you can use Oracle Internet Directory’s authorization framework when identity data resides outside of Oracle Internet Directory.

Server chaining does not replace Oracle Directory Integration Platform. Rather, it offers complementary functionality to Oracle Directory Integration Platform.

Server chaining is different from a virtual directory. A virtual directory, such as Oracle Virtual Directory, is a flexible virtualization layer between multiple identity repositories and applications. It offers complementary services to identity synchronization and directory servers. With a virtual directory, organizations can create consolidated, logical or virtual views of data that may span multiple directories and databases.

Server chaining is a simpler, more flexible solution, embedded in Oracle Internet Directory server, and particular suited to OracleAS Single Sign-On and Enterprise User Security customers. It is easier to administer and upgrade. It also provides Oracle Internet Directory’s authorization framework without extra configuration steps.

This chapter contains the following topics:

- Supported External Servers
- Integrated Oracle Products
- Supported Operations
- Server Chaining with Replication
- Configuring Server Chaining
- Server Chaining Configuration Entries

**Supported External Servers**

Oracle Internet Directory server chaining supports the following external servers:

- Microsoft Active Directory
- Sun Java System Directory Server, formerly known as SunONE iPlanet

An implementation of Oracle Internet Directory can connect with one Active Directory server, one Sun Java System Directory Server, or with both.
Integrated Oracle Products

The following products have been integrated with Oracle Internet Directory server chaining:

- Oracle Application Server Single Sign-On
- Enterprise User Security

When server chaining is enabled, a user from the external directory can:

- Log in through OracleAS Single Sign-On

Oracle Internet Directory server chaining enables you to implement Enterprise User Security without synchronizing identity data with Oracle Internet Directory. Your identity data remains in the external repository and Oracle Internet Directory's data store contains only server chaining metadata.


Supported Operations

Server chaining supports the following operations:

- Bind
- Compare
- Modify
- Search

The compare, modify, and search operations can be enabled or disabled by setting configuration parameters.

When an Oracle Internet Directory client application issues an LDAP search request, Oracle Internet Directory integrates the search results from its own data and the external directories.

When an Oracle Internet Directory client application issues an LDAP bind, compare, or modify request, Oracle Internet Directory redirects the request to the external directory.

In 10g (10.1.4.0.1), the compare operation is only supported for the userpassword attribute.

In 10g (10.1.4.0.1), attribute modification is supported in two cases:

- The external attribute has the same name as the Oracle Internet Directory attribute. This is true for most standard LDAP attributes.
- The external attribute is mapped to an Oracle Internet Directory attribute, and neither the external nor the Oracle Internet Directory attribute is an operational attribute.
Server Chaining with Replication

If you use server chaining in a replication environment, set it up on all nodes so that the entries remain consistent across nodes. Configure server chaining so that the mapped external directories are the same for all the replicated nodes.

Configuring Server Chaining

Oracle Internet Directory is shipped with disabled sample server chaining entries. For Active Directory, the DN for the server chaining entry is

\texttt{cn=oidscad,cn=OID Server Chaining,cn=subconfigsubentry}

For Sun Java System Directory Server, the entry DN is

\texttt{cn=oidsiciplanet,cn=OID Server Chaining,cn=subconfigsubentry}

You configure server chaining by customizing these entries for your environment and enabling them. You can do this either from the command line or by using Oracle Directory Manager.

This section contains the following topics:

- Configuring Server Chaining from the Command Line
- Configuring Server Chaining by Using Oracle Directory Manager
- Requirements for User and Group Containers
- Attribute Mapping

Configuring Server Chaining from the Command Line

Perform the following steps to configure server chaining from the command line:

1. Create an LDIF file to manually add the user and group containers. To determine the DNs for these containers, see the section Requirements for User and Group Containers on page 28-4. For example, if your user search base is \texttt{cn=users,dc=us,dc=oracle,dc=com}, and the group search base is \texttt{cn=groups,dc=us,dc=oracle,dc=com}, then you would use the following entries in your LDIF file:

\begin{verbatim}
  dn: cn=AD,cn=users,dc=us,dc=oracle,dc=com
  cn: AD
  objectclass: orclcontainer
  objectclass: top

  dn: cn=iPlanet,cn=users,dc=us,dc=oracle,dc=com
  cn: iPlanet
  objectclass: orclcontainer
  objectclass: top

  dn: cn=AD,cn=groups,dc=us,dc=oracle,dc=com
  cn: AD
  objectclass: orclcontainer
  objectclass: top
\end{verbatim}

\textbf{Note:} You cannot modify an Active Directory user password from Oracle Internet Directory through server chaining.
Configuring Server Chaining

1. DN of the directory:
   \[ dn: \text{cn=iPlanet,dc=us,dc=oracle,dc=com} \]
   \[ cn: \text{iPlanet} \]
   \[ objectclass: \text{orclcontainer} \]
   \[ objectclass: \text{top} \]

2. Use `ldapadd` and the LDIF file you just created to add the entries.
   \[
   \text{ldapadd -p port -h host -D "binddn" -w password -v -f container_ldif_file_name}
   \]

3. Create another LDIF file to modify and enable the server chaining configuration entries. For example LDIF files, see "Active Directory Example" on page 28-7 and "Sun Java System Directory Server (iPlanet) Example" on page 28-8. A table of attributes is provided in "Server Chaining Configuration Entries" on page 28-5. Attribute mapping is explained in Attribute Mapping on page 28-5.

4. Modify the server chaining configuration entries using the `ldapmodify` command and the LDIF file you just created. Use a command line of the form:
   \[
   \text{ldapmodify -p port -h host -D "binddn" -w password -v -f entry_ldif_file_name}
   \]

Configuring Server Chaining by Using Oracle Directory Manager

Oracle Directory Manager provides a convenient interface for modifying the Oracle Internet Directory server chaining configuration entries. To configure server chaining by using Oracle Directory Manager, perform the following steps:

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

2. Select Server Chaining. The Server Chaining Management window appears in the right pane.

3. To modify a server chaining configuration entry, select it, then choose Edit. The Server Chaining Management Window for iPlanet (Sun Java System Directory Server) or Active directory appears.

4. Click Apply to enable the configuration changes.

See Table A–52 on page A-35 for the online help for descriptions of the fields in the Server Chaining Management Window. Most of the fields correspond with attributes described in Configuration Entry Attributes on page 28-6.

The next section, Requirements for User and Group Containers, provides additional information about the External User Container and External Group Container fields. Attribute mapping is explained in Attribute Mapping on page 28-5.

Requirements for User and Group Containers

The target user and group containers must be under the Oracle Internet Directory search base in order to work with OracleAS Single Sign-On. Use the container names `cn=AD` for Active Directory and `cn=iPlanet` for Sun Java System Directory Server (iPlanet). For example, if your user search base is:

\[ \text{cn=users,dc=us,dc=oracle,dc=com} \]

you would use

\[ \text{cn=AD,cn=users,dc=us,dc=oracle,dc=com} \]

as the target user container for the Active Directory users or
as the target user container for the Sun Java System Directory Server users. Similarly, if
your group search base is:

cn=groups,dc=us,dc=oracle,dc=com

you would use

cn=AD,cn=groups,dc=us,dc=oracle,dc=com

as the target container for the Active Directory or

as the target container for the Sun Java System Directory Server groups.

The target user and group containers exist only for the external directories. All the
users and groups that appear under these nodes are populated by the external
directories. Do not add entries under these containers directly from Oracle Internet
Directory.

**Attribute Mapping**

If an attribute in an external directory and an Oracle Internet Directory attribute are
the same, then no mapping is required. Server chaining performs some attribute
mapping by default. The default mapping list is as follows:

**Table 28–1 Default Attribute Mapping to Active Directory**

<table>
<thead>
<tr>
<th>Oracle Internet Directory Attribute</th>
<th>Active Directory Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>orclguid</td>
<td>objectGUID</td>
</tr>
<tr>
<td>uid</td>
<td>name</td>
</tr>
<tr>
<td>orclsamaccountname</td>
<td>samaccountname</td>
</tr>
<tr>
<td>krbprincipalname</td>
<td>userprincipalname</td>
</tr>
</tbody>
</table>

**Table 28–2 Default Attribute Mapping to Sun Java System Directory Server**

<table>
<thead>
<tr>
<th>Oracle Internet Directory Attribute</th>
<th>Sun Java System Directory Server Attribute</th>
</tr>
</thead>
<tbody>
<tr>
<td>orclguid</td>
<td>nsuniqueid</td>
</tr>
<tr>
<td>authpassword</td>
<td>userpassword</td>
</tr>
<tr>
<td>krbprincipalname</td>
<td>mail</td>
</tr>
</tbody>
</table>

The following objects cannot be mapped:

- Operational attributes
- Object classes
- Oracle Internet Directory- specific attributes. These attributes typically have names
  starting with orcl.

**Server Chaining Configuration Entries**

This section contains the following topics:
- Configuration Entry Attributes
- Active Directory Example
- Sun Java System Directory Server (iPlanet) Example

**Configuration Entry Attributes**

Table 28–3 lists the configuration entry attributes for server chaining.

**Table 28–3  Configuration Entry Attributes for Server Chaining**

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>orclOIDSCExtHost</td>
<td>Yes</td>
<td>The host name of the external directory host. This is a single value attribute.</td>
</tr>
<tr>
<td>orclOIDSCExtPort</td>
<td>Yes</td>
<td>The port number of the external directory host. This is a single value attribute. The default value is 389.</td>
</tr>
<tr>
<td>orclOIDSCExtDN</td>
<td>Yes</td>
<td>The DN in the external directory. Server chaining will bind against the external directory using this identity to perform search and modify operations. This identity must have sufficient privilege to perform the operation. This is a single value attribute.</td>
</tr>
<tr>
<td>orclOIDSCExtPassword</td>
<td>Yes</td>
<td>The password for the DN of the external directory. This is a single value attribute. Be sure to enable privacy mode to ensure that users cannot retrieve this attribute in clear text. See &quot;Privacy of Retrieved Sensitive Attributes&quot; on page 16-2.</td>
</tr>
<tr>
<td>orclOIDSCExtUserContainer</td>
<td>Yes</td>
<td>The user container in the external directory from which to perform the user search operation. This is a single value attribute.</td>
</tr>
<tr>
<td>orclOIDSCExtGroupContainer</td>
<td>No</td>
<td>The group container in the external directory from which to perform the group search operation. This is a single value attribute. This attribute is optional if the external user container and the external group container are the same. In this case the group search operations are performed on the external user container.</td>
</tr>
<tr>
<td>orclOIDSCTargetUserContainer</td>
<td>Yes</td>
<td>The user container in Oracle Internet Directory in which the external users reside. For more information, see &quot;Requirements for User and Group Containers&quot; on page 28-4.</td>
</tr>
<tr>
<td>orclOIDSCTargetGroupContainer</td>
<td>Yes</td>
<td>The group container in Oracle Internet Directory in which the external groups reside. For more information, see &quot;Requirements for User and Group Containers&quot; on page 28-4.</td>
</tr>
</tbody>
</table>
The following example shows server chaining configured to use the Active Directory server `dlin-pc9.us.oracle.com`, port 389, as its external directory store. All the attributes are explained in Table 28–3 on page 28-6.

```
cn=oidscad,cn=OID Server Chaining,cn=subconfigsubentry
orclOIDSCExtHost: dlin-pc9.us.oracle.com
orclOIDSCExtPort: 389
orclOIDSCExtDN: cn=administrator,cn=users,dc=oidvd,dc=com
orclOIDSCExtPassword: ********
orclOIDSCExtUserContainer: cn=users,dc=oidvd,dc=com
orclOIDSCTargetUserContainer: cn=AD,cn=users,dc=us,dc=oracle,dc=com
orclOIDSCTargetGroupContainer: cn=AD,cn=groups,dc=us,dc=oracle,dc=com
orclOIDSCExtSearchEnabled: 1
orclOIDSCExtModifyEnabled: 1
orclOIDSCExtAuthEnabled: 1
orclOIDSCAttrMapping;description: title
```

For more information, see "Attribute Mapping" on page 28-5.

**Active Directory Example**

The following example is the LDIF file used to modify the configuration entry:

```
dn: cn=oidscad,cn=OID Server Chaining,cn=subconfigsubentry
changetype: modify
replace: orcloidscextdn
orcloidscextdn: administrator@dlin.net
replace: orcloidscextpassword
orcloidscextpassword: welcome1
replace: orcloidscexthost
orcloidscexthost: dlin-pc9.us.oracle.com
replace: orcloidscextport
orcloidscextport: 389
replace: orcloidsctargetusercontainer
orcloidsctargetusercontainer: cn=ad,cn=users,dc=us,dc=oracle,dc=com
```

Table 28–3 (Cont.) Configuration Entry Attributes for Server Chaining

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Required</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>orclOIDSCAttrMapping</td>
<td>No</td>
<td>Specifies each attribute mapping between the external directory and Oracle Internet Directory. For example, to map the <code>eMail</code> attribute from Active Directory to the <code>mail</code> attribute in Oracle Internet Directory, set this attribute to: <code>orclOIDSCAttrMapping;mail:eMail</code></td>
</tr>
</tbody>
</table>
Server Chaining Configuration Entries

replace: orcloidsctargetgroupcontainer
orcloidsctargetgroupcontainer: cn=*ad*, cn=*groups*, dc=*us*, dc=*oracle*, dc=com

replace: orcloidscextusercontainer
orcloidscextusercontainer: cn=*users*, dc=*dlin*, dc=*net*

replace: orcloidscextgroupcontainer
orcloidscextgroupcontainer: cn=*users*, dc=*dlin*, dc=*net*

replace: orcloidscextsearchenabled
orcloidscextsearchenabled: 1

replace: orcloidscextmodifyenabled
orcloidscextmodifyenabled: 1

replace: orcloidscextauthenabled
orcloidscextauthenabled: 1

Sun Java System Directory Server (iPlanet) Example

The following example shows server chaining configured to use the Sun Java System Directory Server dlin-pc10.us.oracle.com, port 10389, as its external directory store. All the attributes are explained in Table 28–3 on page 28-6.

cn=oidsciplanet,cn=OID Server Chaining, cn=subconfigsubentry
orclOIDSCExtHost: dlin-pc10.us.oracle.com
orclOIDSCExtPort: 10389
orclOIDSCExtDN: cn=directory manager
orclOIDSCExtPassword: ******
orclOIDSCExtUserContainer: ou=people, dc=acme, dc=com
orclOIDSCExtGroupContainer: ou=groups, dc=acme, dc=com
orclOIDSCTargetUserContainer: cn=iPlanet, cn=users, dc=us, dc=oracle, dc=com
orclOIDSCTargetGroupContainer: cn=iPlanet, cn=groups, dc=us, dc=oracle, dc=com
orclOIDSCExtSearchEnabled: 1
orclOIDSCExtModifyEnabled: 1
orclOIDSCExtAuthEnabled: 1

The following example is the LDIF file used to modify the configuration entry:

dn: cn=oidsciplanet, nc=OID Server Chaining, nc=subconfigsubentry
changetype: modify
replace: orcloidscextdn
orcloidscextdn: cn=directory manager

replace: orcloidscextpassword
orcloidscextpassword: welcome1

replace: orcloidscexthost
orcloidscexthost: dlin-pc10.us.oracle.com

replace: orcloidscextport
orcloidscextport: 10389

replace: orcloidsctargetusercontainer
orcloidsctargetusercontainer: cn=iPlanet, cn=users, dc=us, dc=oracle, dc=com

replace: orcloidsctargetgroupcontainer
orcloidsctargetgroupcontainer: cn=iPlanet, cn=groups, dc=us, dc=oracle, dc=com
Debugging Server Chaining

To debug server chaining, perform the following steps:

1. Set the Oracle Internet Directory server debug logging level, as described in "Setting Debug Logging Levels" on page 14-5. Use the logging level value 402653184. This value enables logging of all messages related to the Java plug-in framework.

2. Modify the Oracle Internet Directory server chaining debugging settings. For both cn=oidscad,cn=oid server chaining,cn=subconfigsubentry and cn=oidsciplanet,cn=oid server chaining, cn=subconfigsubentry. set the attribute orcloidscDebugEnabled to 1.

For example, to set orcloidscDebugEnabled to 1 in cn=oidscad,cn=oid server chaining,cn=subconfigsubentry, you would type:

```
$ORACLE_HOME/bin/ldapmodify -h host -p port -D cn=orcladmin -w orcladminpwd -

dn: cn=oidscad,cn=oid server chaining,cn=subconfigsubentry
changetype: modify
replace: orcloidscDebugEnabled
orcloidscDebugEnabled: 1
```

See Also: The Java Plug-in Debugging and Logging section in Oracle Identity Management Application Developer’s Guide.
This part provides detailed discussions of replication and high availability and how to plan and manage them. It contains these chapters:

- Chapter 29, "Oracle Internet Directory Replication Concepts"
- Chapter 30, "Oracle Internet Directory Replication Installation and Configuration"
- Chapter 31, "Oracle Internet Directory Replication Monitoring and Management"
This chapter provides an introduction to Oracle Internet Directory replication. Replication is the process of copying and maintaining the same naming contexts on multiple directory servers. It can improve performance by providing more servers to handle queries and by bringing the data closer to the client. It improves reliability by eliminating risks associated with a single point of failure.

This chapter contains the following topics:

- Replication Concepts
- Directory Replication Groups
- Replication Configuration Objects in the Directory
- Replication Security
- Change Logs in Directory Replication
- Oracle Database Advanced Replication
- LDAP-Based Replication
- Conflict Resolution in Oracle Replication
- Replication Failover
- Included and Excluded Naming Contexts in Partial Replication
- Oracle Database Advanced Replication Filtering
- LDAP Replication Filtering

**Replication Concepts**

This section briefly introduces some basic concepts. Later sections describe these concepts in more detail.

This section contains the following topics:

- Content to be Replicated: Full or Partial
- Direction: One-Way or Two-Way
- Transport Mechanism: Advanced Database Replication or LDAP
- Directory Replication Group (DRG) Types
Content to be Replicated: Full or Partial

One decision you must make when setting up replication is how much of the DIT to replicate from one node to another. The choices are:

<table>
<thead>
<tr>
<th>Table 29–1 Full or Partial Replication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Content to Replicate</td>
</tr>
<tr>
<td>Full</td>
</tr>
<tr>
<td>Partial</td>
</tr>
</tbody>
</table>

Full replication involves propagating the entire DIT to another node. This type of replication ensures the high availability of the entire directory. You can also use it to distribute operations on the entire directory among different nodes. Full replication can be based on either Oracle Database Advanced Replication or LDAP.

Partial replication enables you to propagate one or more subtrees, rather than the entire DIT, to another node. Decentralizing a directory in this way enables you to balance the workload between servers and build a highly available distributed directory, complete with fault tolerance and failover. You can configure partial replication by using the Replication Environment Management Tool. Partial replication is most often LDAP-based.

Figure 29–1 shows an example of partial replication.

Direction: One-Way or Two-Way

The direction of replication is a property of the replication agreement you set up between nodes. It can be:

<table>
<thead>
<tr>
<th>Table 29–2 Direction of Replication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Direction</td>
</tr>
<tr>
<td>One-Way</td>
</tr>
</tbody>
</table>
Sometimes the terms read-only and read-write are used to describe direction of replication. In a one-way replication agreement, the consumer node is said to be read-only. That is, you cannot propagate changes to other nodes by writing to that node. In a two-way replication agreement, both nodes are considered to be read-write. Another term for read-write is updateable.

**Transport Mechanism: Advanced Database Replication or LDAP**

Oracle Internet Directory supports two protocols you can use for replicating data from one node to another. The protocols are:

<table>
<thead>
<tr>
<th>Transport Mechanism</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Oracle Advanced Database Replication</td>
<td>Uses the replication feature of Oracle Database. Also called Advanced Replication.</td>
</tr>
<tr>
<td>LDAP</td>
<td>Uses the industry-standard Lightweight Directory Access Protocol Version 3</td>
</tr>
</tbody>
</table>

Advanced Replication is usually peer-to-peer. LDAP replication can be configured as either one-way or two-way.

**Directory Replication Group (DRG) Types**

The directory servers that participate in the replication of a given naming context form a directory replication group (DRG). The relationship among the directory servers in a DRG is represented on each node by a special directory entry called a replication agreement. A replication agreement can be either one-way or two-way.

Each copy of a naming context contained within a server is called a replica. A server that sends the updates made to it to other servers is known as a supplier. A server that accepts those changes is called a consumer. A server can be both a supplier and a consumer.

A directory replication group can be single-master, multimaster, or fan-out as described in Table 29–4.

**Table 29–4 Types of Directory Replication Groups**

<table>
<thead>
<tr>
<th>Group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single-master</td>
<td>Has only one supplier replicating changes to one or more consumers. Clients can update only the master node, and can only read data on any of the consumers. This type of group typically uses LDAP. It is also possible to configure Advanced Replication as single-master, by switching all nodes in a group except one to read-only mode.</td>
</tr>
</tbody>
</table>
In a directory replication group, the protocol for transferring data can be based on either Oracle Database Advanced Replication or LDAP.

Table 29–5 Types of Data Transfer Between Nodes in a Directory Replication Group

<table>
<thead>
<tr>
<th>Feature</th>
<th>LDAP-Based Replication</th>
<th>Advanced Replication</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change propagation</td>
<td>Change propagation from supplier to consumer uses LDAP</td>
<td>Change propagation from supplier to consumer uses Advanced Replication</td>
</tr>
<tr>
<td>Content replicated</td>
<td>Full replica</td>
<td>Full replica (usually)</td>
</tr>
<tr>
<td></td>
<td>Partial replica</td>
<td></td>
</tr>
<tr>
<td>Direction of replication</td>
<td>One-way</td>
<td>Peer-to-peer</td>
</tr>
<tr>
<td></td>
<td>Two-way</td>
<td></td>
</tr>
<tr>
<td>Deployment Configuration</td>
<td>Single-master replication</td>
<td>Multimaster replication</td>
</tr>
<tr>
<td></td>
<td>Fan-out replication</td>
<td>Single-master replication, by switching all masters in a multimaster configuration except one to read-only mode.</td>
</tr>
</tbody>
</table>
Single-Master Replication Groups

A single-master replication group has only one supplier replica supplying changes to one or more consumers. All replication agreements are one-way, from the supplier to the consumer. Clients can update only the master replica, and can only read data on any of the consumers.

Figure 29–2, "Example of Single-Master Replication" shows a single-master replication environment.

![Figure 29–2 Example of Single-Master Replication](image)

In Figure 29–2, each bullet represents a node of Oracle Internet Directory. Node A is a supplier that replicates consumer nodes B and C. Node A is read/write, and Nodes B and C are read-only. The data transfer protocol is LDAP.

Multimaster Replication Groups

A multimaster replication group, also called a peer-to-peer replication group, has two or more nodes acting as equals to manage groups of replicated data. In a multimaster replication group, each directory server is both a supplier and a consumer of changes, and the entire directory is replicated on each node.

The example in Figure 29–3 shows three nodes—A, B, and C—that update each other in a multimaster replication group. Replication between nodes is two-way.

![Figure 29–3 Example of Multimaster Replication](image)
In Figure 29–3, all replication is two-way, and the data transfer protocol is based on Advanced Replication.

**Note:** Multimaster replication is the only replication mechanism supported in Oracle Application Server Single Sign-On as described in the section "Configuring Oracle Application Server Single Sign-On for Replication" in the chapter on high availability in the Oracle Application Server Single Sign-On Administrator’s Guide

**See Also:** "Oracle Database Advanced Replication" on page 29-20 for more information about multimaster replication

**Fan-Out Replication Groups**

A fan-out replication group, also called a point-to-point replication group, has a supplier replicating directly to a consumer. That consumer can then supply the same data to one or more other consumers. The replication can be either full or partial and either one-way or two-way.

Figure 29–4 shows a fan-out replication environment.

*Figure 29–4  Example of Fan-Out Replication*

In Figure 29–4, supplier A replicates to two consumers, B and C. Consumer node B contains a partial replica of A, whereas consumer node C contains a full replica of A. Each of these nodes, in turn, serves as a supplier that replicates data to two other consumers: Node B partially replicates to nodes D and E, and node C fully replicates to nodes F and G. Nodes D and F are read-only.

In fan-out replication, nodes transfer data by using LDAP.

**Types of Directory Replication Compared**

Table 29–6 compares multimaster, single-master, and fan-out replication.
Multimaster Replication with Fan-Out

Oracle Internet Directory enables any node in a multimaster replication group to also participate in an LDAP-based replication agreement. The node it connects to using LDAP can, in turn, supply data to other nodes in a fan-out configuration. Within the multimaster replication agreement, data transfer between the nodes occurs by way of Oracle Database Advanced Replication. Within the fan-out replication agreement, data transfer from supplier to consumer occurs by way of LDAP. The LDAP replication agreements can be either one-way or two-way.

**Note:** If an LDAP-based replica is read/write, then changes on this node propagate to consumers, but not to suppliers.

Figure 29–5 shows an example of multimaster replication used in conjunction with fan-out replication.
Figure 29–5  Example of Multimaster Replication with Fan-Out

In the example in Figure 29–5, nodes A, B, and C form a multimaster replication group. They transfer data among them by using Advanced Replication.

Node B supplies changes to Node D, a replica of the entire directory. Node D, in turn, supplies changes to Nodes F and G by using LDAP-based replication. Both Nodes F and G are replicas of the entire directory. Similarly, Node E is a full replica of Node C. Node E, in turn supplies changes to Node H, a replica of the entire directory, and Node I, a partial replica, by using LDAP-based replication. Nodes F and H are read-only.

See Also:  "LDAP-Based Replication" on page 29-23 for more information about fan-out replication

Replication Configuration Objects in the Directory

This section describes the objects in the directory that contain replication configuration information. It contains these topics:

- The Replication Configuration Container
- The Replica Subentry
- The Replication Agreement Entry
- The Replication Naming Context Container Entry
- The Replication Naming Context Object Entry
- Examples of Replication Configuration Objects in the Directory
The Replication Configuration Container

All replication information for a node resides in the container `cn=replication configuration` located at the root DSE. This entry resides on each node in a DRG. The following is a sample replication configuration container entry:

```
dn: cn=replication configuration
orclaci: access to entry by * (browse)
orclaci: access to attr=(*) by * (search,read)
orclnormdn: cn=replication configuration
cn: replication configuration
description: Replication agreement Container object
objectclass: top
objectclass: orclcontainerOC
```

The Replica Subentry

This subentry is created at installation under the replication configuration container. It contains attributes that identify and define the characteristics of the node it represents. Table 29–7 describes the attributes of the replica subentry.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OrclReplicaID</td>
<td>Naming attribute for the replica subentry. Its value is unique to each directory server node that is initialized at installation. The value of this attribute, assigned during installation, is unique to each directory node, and matches that of the orclreplicaID attribute at the root DSE. You cannot modify this value.</td>
</tr>
<tr>
<td>orclReplicaURI</td>
<td>Contains information in ldapURI format that can be used to open a connection to this replica.</td>
</tr>
<tr>
<td>orclReplicaSecondaryURI</td>
<td>Contains the set of ldapURI format addresses that can be used if the orclReplicaURI values cannot be used.</td>
</tr>
<tr>
<td>orclReplicaType</td>
<td>Defines the type of replica such as read-only or read/write. Possible values are:</td>
</tr>
<tr>
<td></td>
<td>■ 0 (Read/Write)</td>
</tr>
<tr>
<td></td>
<td>■ 1 (Read-Only)</td>
</tr>
<tr>
<td>orclReplicaState</td>
<td>Defines the state of the replica. See Appendix H, &quot;LDAP Replica States&quot; for more information.</td>
</tr>
</tbody>
</table>

Table 29–7  Attributes of the Replica Subentry

In Figure 29–8 on page 29-18, a replica subentry is represented by `orclReplicaID=UID_of_node_D,cn=replication configuration`. The following is a sample replica subentry:

```
dn: orclreplicaid=myhost1_repl1,cn=replication configuration
objectclass: top
objectclass: orclreplicasubentry
orclreplicaid: myhost1_repl1
orclreplicauri: ldap://myhost1:3060/
orclreplicasecondaryuri: ldap://myhost1.mycompany.com:3060/
orclreplicastate: 1
```
See Also: "Replication Schema Elements" in Oracle Identity Management User Reference for descriptions of the attributes of the replica subentry.

The Replication Agreement Entry

This entry contains attributes that define the replication agreement between the two or more nodes and is associated with the orclReplAgreementEntry objectclass. There are two kinds of agreement:

1. Oracle Database Advanced Replication Agreement. The replication agreement for Oracle Database Advanced Replication nodes resides under the replication configuration entry. For example: In Figure 29–7, an Oracle Database Advanced Replication agreement entry is represented by orclagreementID=000001.

2. LDAP-based replication agreement. The replication agreement for LDAP nodes resides under the supplier’s replica subentry. For example, in Figure 29–8, an LDAP-based replication agreement entry is represented by orclagreementID=000003, orclReplicaID=UID_of_node_D, cn=replication configuration.

Replication Agreement Entry Attributes

Table 29–8 shows the attributes in the replication agreement.

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>orclagreementID</td>
<td>Naming attribute for the replication agreement entry. You cannot modify this attribute.</td>
</tr>
<tr>
<td>OrclReplicaDN</td>
<td>For LDAP-based replication only. Specifies the DN of the replica to identify a consumer in the replication agreement. You cannot modify this attribute.</td>
</tr>
<tr>
<td>OrclReplicationProtocol</td>
<td>Define the replication protocol for change propagation to replica. Values:</td>
</tr>
<tr>
<td></td>
<td>- ODS_ASR_1.0 specifies Advanced Replication-based protocol</td>
</tr>
<tr>
<td></td>
<td>- ODS_LDAP_1.0 specifies LDAP-based replication</td>
</tr>
<tr>
<td>OrclDirReplGroupDSAs</td>
<td>For Advanced Replication-based groups, the orclreplicaid values of all the nodes in this replication group. This list must be identical on all nodes in the group. You can modify this attribute. This attribute is not applicable for LDAP-based agreement.</td>
</tr>
<tr>
<td>OrclUpdateSchedule</td>
<td>Replication update interval for new changes and those being retried. The value is in minutes. You can modify this attribute.</td>
</tr>
<tr>
<td>OrclHIQSchedule</td>
<td>The interval, in minutes, at which the directory replication server repeats the change application process. You can modify this attribute.</td>
</tr>
<tr>
<td>OrclLDAPConnKeepAlive</td>
<td>Attribute determining whether the connections from the directory replication server to the directory server is kept active or established every time the changelog processing is done based on various schedules. You can modify this field.</td>
</tr>
</tbody>
</table>
Advanced Replication Agreements

For Advanced Replication, the replication agreement on each node lists all of the nodes in the group. It is identical on each node except for local options such as partitioned naming contexts on the local directory server.

The entry for this kind of replication agreement resides immediately below the cn=replication configuration container entry. For example, the DN of such an agreement can look like this: orclagreementID=000001, cn=replication configuration.
LDAP Replication Agreements

For LDAP-based replication, there are separate replication agreements for each supplier-consumer relationship. For one-way replication, there is a single, one-way replication agreement.

The entry for an LDAP-based replication agreement resides immediately below the replica subentry of the node that serves as the supplier. Thus, the DN of the replication agreement as found on a supplier node is:

```
orclagreementID=unique_identifier_of_the_replication_agreement,
orclReplicaID=unique_identifier_of_supplier_node, cn=replication configuration
```

Similarly, the DN of the replication agreement as found on a consumer node is:

```
orclagreementID=unique_identifier_of_the_replication_agreement,
orclReplicaID=unique_identifier_of_supplier_node, cn=replication configuration
```

In a fan-out replication agreement, you can tell which node the agreement entry is associated with by looking at its parent. For example, look at the following replication agreement entry.

```
orclagreementID=000002, orclReplicaID=node_A, cn=replication configuration
```

In this example, you can determine that the replication agreement represented by orclagreementID=000002 is associated with node A. This is because the parent of orclagreementID=000002 is orclReplicaID=node_A.

---

**Note:** The container entry cn=replication configuration is replicated on all nodes, but may not be identical on all nodes.

---

**Note:** The orclreplicadn attribute of an LDAP-based replication agreement specifies the associated consumer node.

---

Two-Way LDAP Replication Agreements

For two-way replication, there can be either a single, two-way replication agreement or two one-way agreements for each supplier-consumer relationship. The following is a sample two-way replication agreement entry:

```
dn: orclagreementid=000002, orclreplicaid=stadd58, cn=replication configuration
orclagreementid: 000002
orclreplicationprotocol: ODS_LDAP_1.0
orclreplicadn: orclreplicaid=stadd57, cn=replication configuration
orclldapconnkeepalive: 1
orclagreementtype: 1
orclreplicationid: 000002
orcllastappliedchangenumber;transport$stadd57$stadd58: 106
orcllastappliedchangenumber;transport$stadd58$stadd57: 2421
orcllastappliedchangenumber;apply$stadd57$stadd58: 106
orcllastappliedchangenumber;apply$stadd58$stadd57: 2421
orclupdateschedule: 0
orclhiqschedule: 1
objectclass: orclReplAgreementEntry
objectclass: top
```
The Replication Naming Context Container Entry

This entry contains all the LDAP naming context objects.

This entry has the RDN cn=replication namecontext, and it is created below the orclagreementID entry during replication configuration. The following is a sample replication naming context container entry:

dn: cn=replication namecontext,orclagreementid=000002,
orclreplicaid=myhost1_repl1,cn=replication configuration
objectclass: top
objectclass: orclcontainerOC
cn: replication namecontext

The Replication Naming Context Object Entry

This entry contains all the LDAP naming context objects. These objects specify the replication filtering policy, that is, what to include in or exclude from replication to an LDAP-based partial replica.

Note: You cannot include naming contexts or exclude attributes in Advanced Replication-based agreements. Advanced Replication agreements use the base attribute orclexcludednamingcontexts described in Table 29–8.

This entry is created below the naming context container entry during replication configuration. It is configurable. For example, in Figure 29–8 on page 29-18, the replication naming context object is cn=namingcontext001,cn=replication namecontext,orclagreementID=000003,orclReplicaID=UID_of_node_D,cn=replication configuration

Table 29–9 Attributes of the Replication Naming Context Entry

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>orclincludednamingcontexts</td>
<td>The root of the naming context to be replicated. This attribute has subtypes that specify the replication direction in which the naming context should be included. The format is: orclincludednamingcontexts ; supplier_replicaID$consumer_replicaID: DN</td>
</tr>
<tr>
<td></td>
<td>This is a single valued attribute. For each naming context object, you can specify only one unique subtree in each direction. In partial replication, except for subtrees listed in the orclexcludednamingcontexts attribute, all subtrees in the specified included naming context are replicated. You can modify this attribute.</td>
</tr>
</tbody>
</table>
The following is a sample replication naming context object entry:

dn: cn=namectx001,
  cn=replication namecontext,
  orclagreementid=unique_identifier_of_the_replication_agreement,
  orclreplicaid=replica_id_of_node_A,
  cn=replication configuration
  orclincludednamingcontexts: cn=mycompany
  orclincludednamingcontexts; replica_id_of_node_A$ replica_id_of_node_B : c=us,cn=mycompany
  orclexcludedattributes; replica_id_of_node_B$ replica_id_of_node_A : userPassword

The example specifies the following replication filtering:

- The naming context `cn=mycompany` is included for replication in both directions for node A and node B.
- The naming context `c=us,cn=mycompany` is excluded for replication from node A to node B only.
- The `userPassword` attribute is excluded for replication from node B to node A

See Also:

- "Examples of LDAP Replication Filtering" on page 29-31
- "Replication Schema Elements" in Oracle Identity Management User Reference

### Directory Replication Server Configuration Parameters

Table 29–10 lists and describes the attributes of the replication server configuration set entry, which has the following DN:

cn=configset0,cn=osdrep1d,cn=subconfigsubentry
Examples of Replication Configuration Objects in the Directory

The examples of replication objects in this section rely on the replication environment shown in Figure 29–6.

---

Table 29–10 Directory Replication Server Configuration Parameters

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>modifyTimestamp</td>
<td>Time of entry creation or modification. You cannot modify this parameter.</td>
</tr>
<tr>
<td>modifiersName</td>
<td>Name of person creating or modifying the entry. You cannot modify this parameter.</td>
</tr>
<tr>
<td>orclChangeRetryCount</td>
<td>Single-valued attribute. The number of processing retry attempts for a change-entry before being moved to the human intervention queue. The value for this parameter must be equal to or greater than 1 (one). The default is 10. You can modify this parameter.</td>
</tr>
<tr>
<td>orclThreadsPerSupplier</td>
<td>Number of worker threads spawned at each supplier for transporting or applying change logs. This parameter has two subtypes. The format is: orclthreadspersupplier; work_type: Number The possible values of work_type are: transport apply The default is one worker thread for transport and five worker threads for apply. You can modify this parameter.</td>
</tr>
</tbody>
</table>
In Figure 29–6, nodes A, B, and C form a multimaster replication group. Node C replicates to a fourth node, D, which, in turn, fans out to Node E.

The replication agreements in this environment are as follows:

- Node A has one replication agreement representing its multimaster relationship with nodes B and C.
- Node B has two replication agreements, the first representing its multimaster relationship with nodes A and C, the second representing its relationship to node F. The replication agreement between B and F is two-way.
- Node C has two replication agreements, the first representing its multimaster relationship with nodes A and B, the second representing its relationship to node D. This is a one-way replication agreement in which C serves as the supplier and node D is the consumer.
- Node D has two replication agreements. Both of its replication agreements are one-way. One represents its relationship to the supplier node C, from which it consumes changes, the other represents its relationship to consumer node E for which it is the supplier.
- Node E has a one-way replication agreement with Node D. Node E is the consumer.
- Node F has two replication agreements, one representing its relationship to the node B, the other representing its relationship to node G. Both are two-way replication agreements.
- Node G has a one-way replication agreement with Node F. Node G is the consumer.
Figure 29–7 shows the replication objects in the DIT that pertain to node C in Figure 29–6 on page 29-16.

For node C, the entry *cn=replication configuration* at the root DSE contains these RDNs:

- *orclagreementID=000001*: The multimaster replication agreement in which node C participates with nodes A and B.
- *orclReplicaID=UID_of_node_C*: Unique identifier of node C that contains information about it.
- *orclagreementID=000002*: Unique identifier of the relationship between supplier node C and consumer node D. You know that, in this case, *orclagreementID=000002* is the replication agreement of the supplier node C because node C is its parent.

This entry contains the attribute *orclreplicaDN*. Its value is the replica entry DN of consumer node D, with which node C has the replication agreement.

- *cn=replication DN*: The bind DN that the directory replication server on node C uses to bind to the directory server.
- *cn=replication namecontext*: Container of information about naming contexts that are included in replication.
- *cn=namingcontext001* and *cn=namingcontext002*: The actual objects that are included in or excluded from replication. In the naming context included for replication, you can specify one or more subtrees to be excluded from replication. In that same included naming context, you can specify particular attributes to be excluded from replication.

Figure 29–8 shows the replication agreement entry in the DIT that pertains to node D in Figure 29–6 on page 29-16.
For node D, the entry `cn=replication configuration` at the root DSE contains these RDNs:

- `orclReplicaID=UID_of_node_D`: Unique identifier of node D and contains information about it.
- `orclagreementID=000003`: Unique identifier of the relationship between supplier node D and consumer node E. You know that, in this case, `orclagreementID=000003` is the replication agreement of the supplier node D because node D is its parent.

This entry contains the attribute `orclreplicaDN`, the value of which is the DN of consumer node E with which node D has the replication agreement.

- `cn=replication DN`: Bind DN that the directory replication server on node D uses to bind to the directory server.
- `cn=replication namecontext`: Container of information about naming contexts that are included in replication.
- `cn=namingcontext001` and `cn=namingcontext002`: Objects specifying naming contexts to be included in replication. In the naming context included in replication, you can specify one or more subtrees or particular attributes to be excluded from replication.

**Replication Security**

This section contains these topics:

- **Authentication and the Directory Replication Server**
- **Secure Sockets Layer (SSL) and Oracle Internet Directory Replication**

**Authentication and the Directory Replication Server**

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

It is important that the directory replication server be properly authenticated before it is allowed access to the directory.

The directory replication server uses a unique identity and a password to authenticate with the directory server. The identity of the directory replication server is of the form
When it starts, the directory replication server reads its identity and password from an Oracle Internet Directory secure wallet, and uses these credentials for authentication. If you want to change the password for the replication bind DN, then you must use the \(-pchgpw\), \(-presetpwd\), or \(-pchgwalpwd\) option of the Replication Environment Management Tool. The wallet for replication identity is located at \$ORACLE_HOME/ldap/admin/oidpwr\_Oracle\_SID.

**See Also:** The `remtool` command-line tool reference in Oracle Identity Management User Reference.

---

**Note:** In earlier releases, the replication server required the directory server to allow anonymous bind. The replication server no longer requires that.

---

Secure Sockets Layer (SSL) and Oracle Internet Directory Replication

You can deploy Oracle Internet Directory replication with or without SSL. Replication automatically detects if the target Oracle Internet Directory instance is running at an SSL port. When the replication server binds to the SSL port of an Oracle Internet Directory instance, it will automatically work on top of the Secure Sockets Layer.

**Note:** In Oracle Internet Directory 10g (10.1.4.0.1), the Oracle directory replication server cannot communicate directly with an SSL-enabled LDAP server that supports two way (mutual) authentication. The replication server startup will fail and hang if the LDAP server is configured for SSL mutual authentication.

To configure LDAP-based replication to use SSL encryption, in the `orclReplicaURI` attribute, which contains the supplier contact information, specify the port number of the SSL port.

To configure Advanced Replication to use SSL encryption, use Oracle Advanced Security.

**See Also:** Oracle Advanced Security Administrator’s Guide for information on how to configure Advanced Replication to use SSL encryption

---

Change Logs in Directory Replication

Oracle Internet Directory records each change as an entry in the change log store. The consumer keeps track of the change number of the last change it applied, and it retrieves from the supplier only those changes with numbers greater than that of the last change it applied.

Each entry in the change log store—that is, each change log object—has a unique change number. The consumer retrieves from the supplier only those changes with numbers greater than that of the last change it applied.

- In an LDAP-based replication agreement, change log processing consists of two phases, transporting the change log and applying the change log. For each LDAP-based agreement, there are two change log processing statuses, one for the
each phase. The directory replication server stores the last change number it transported in the transport subtype of the orlcllastappliedchangenumber attribute of the replication agreement entry. The directory replication server stores the last change number it applied in the apply subtype of the orlcllastappliedchangenumber attribute of the replication agreement entry. The format of the orlcllastappliedchangenumber attribute is shown in Table 29–8, "Attributes of the Replication Agreement Entry" on page 29-10.

- In a replication agreement based on Advanced Replication, the directory replication server stores the last change number it transported in the changenumber attribute of the changestatus entry. The changenumber attribute looks like this:

  changenumber=last_applied_change_number, supplier=supplier_node, consumer=consumer_node

  For example, if the last change a consumer applied had a number of 250, then subsequent changes it retrieves from that supplier would need to have numbers greater than 250.

  Change logs are purged by the garbage collector after they have been consumed by the replication server.

  **See Also:**
  - Chapter 26, "Garbage Collection in Oracle Internet Directory"
  - "Oracle Database Advanced Replication" on page 29-20
  - "LDAP-Based Replication" on page 29-23

### Oracle Database Advanced Replication

This section gives a detailed look at Oracle Database Advanced Replication. This section contains these topics:

- **Features of Oracle Database Advanced Replication**
- **Architecture for Oracle Advanced Database Replication**

  **See Also:**
  - Chapter 31, "Oracle Internet Directory Replication Monitoring and Management"
  - Appendix F, "The Multimaster Replication Process"

### Features of Oracle Database Advanced Replication

In Oracle Database Advanced Replication, the transport of update information between nodes in a replication agreement is managed by Oracle Database Advanced Replication, a store-and-forward transport feature. Advanced Replication enables you to synchronize database tables across two Oracle databases.

**Oracle Database Advanced Replication:**

- Stores local changes and periodically propagates them in batches to consumers. The consumer replication servers apply the remote changes to their own local directory servers, and then purge the applied remote changes from their local stores.
Oracle Database Advanced Replication

- Enables read and update access to directory tables anywhere in the Oracle replication group. Typical Advanced Replication configurations use row-level replication.

- Provides proven network tolerance. Data transfer can be controlled and monitored by Oracle Enterprise Manager 10g Application Server Control Console. Such manageability allows a high degree of flexibility in how the data transfer is scheduled.

---

**Note:** The Oracle Application Server Single Sign-On database schema that resides in the same database as Oracle Internet Directory is also replicated by using Advanced Replication.

---

**See Also:**


- Oracle Database Advanced Replication in the Oracle Database Documentation Library for information about Advanced Replication

### Architecture for Oracle Advanced Database Replication

Typical Oracle Database Advanced Replication configurations use asynchronous data propagation—that is, suppliers write their changes to change logs, and then regularly send batched changes to other consumers. Consumers receive the change log data, then reproduce the changes locally.

When you configure replication, you specify which nodes in a replication group share changes. Regardless of the number of nodes you introduce into the replication environment, the basic architecture for replication remains the same. Local changes are distributed to a remote master site where the replication server, acting as a client, sends commands to the Oracle Internet Directory server that implements them.

*Figure 29–9* and its accompanying text provides an overview of the Oracle Advanced Database Replication process.
The events are as follows:

1. A change request is made on the Oracle Internet Directory server of Replica A.

2. The change is accepted and committed to storage in the Oracle Internet Directory Database.

   2' A new change log is generated for the operation, and stored in the ods_chg_log table, where server = Replica A.

3. The replication server on Replica A queries for new outbound change logs from the local ods_chg_log table. The filter used for the query is:

   ```
   (& (objectclass=changeLogEntry) (servername=ReplicaID_of_A) (changeNumber>= Last_Applied_ChgNum_From_A_TO_A) )
   ```

   A control is also passed along with this search operation with the value of orclreplicationid specified in the processing agreement. The value 1 is reserved for Advanced Replication agreements.

   Last_Applied_ChgNum_From_A_TO_A is the outbound change log processing status.

4. The new change log retrieved from ods_chg_log is copied to local asr_chg_log.

5. The replication server requests Oracle Internet Directory server to update the Change log retry_cnt status properly.

   5' Oracle Internet Directory server updates the retry_cnt of the change log in the ods_chg_log table.

6. The new change log is pushed From Replica A to Replica B through Oracle Advanced Database replication.

7. The replication server on Replica B queries for new inbound change logs from its local asrchg_log table. The filter used is:

   ```
   (& (objectclass=changeLogEntry) (servername=ReplicaID_of_A) (changeNumber>= Last_Applied_ChgNum_From_A_TO_B) )
   ```
8. The replication server applies the new change at Replica B.

9. Oracle Internet Directory server at Replica B accepts and commits the change to storage in the Oracle Database.

9' If the local replica has any LDAP-based consumer replicas, a change log is regenerated in the ods_chg_log table at Replica B, with the following regenerating rules:

   server = server of local replica,
   ReplicaB orclreplicationid/chg_rid = 1
   Modifiersname = Replbind_DN_of_A
   orclreplicatid/chg_rid is set to the value of orclreplicationid of the processing agreement. In this case, because the change is replicated and processed through Advanced Replication, it is set to 1 because 1 is reserved for Advanced Replication.

10. The replication server request Oracle Internet Directory server to update the shadow change log retry_cnt status properly.

10' Oracle Internet Directory server updates the retry_cnt of the shadow change log in the database.

Purging occurs regularly, removing entries that are already applied and those that are dropped as candidate changes. Remote change records in the local change log table are purged by the garbage collection thread if they have been applied locally. Local change records in the local change log table are purged by the garbage collection thread if they have been distributed to all the consumers.

See Also: Chapter 31, "Oracle Internet Directory Replication Monitoring and Management" for information on configuring replication

LDAP-Based Replication

This section gives a more detailed look at LDAP replication. Data flow in LDAP replication consists of two phases, transport and apply.

Figure 29–10 and its accompanying text explain the fan-out replication process.
1. An LDAP change request is made on the Oracle Internet Directory server of Replica A.

2. The change is accepted and committed to Oracle Internet Directory database storage.

   2’. A new change log is generated for the operation and stored in the `ods_chg_log` table, where:
   
   - `server = Replica A`
   - `orclreplicationid/chg_rid = 0` (The value 0 is for user operations)
   - `Modifiersname = user DN of the Modifier (by default)`

3. The replication server on Replica B queries for new inbound change logs from the supplier replica, Replica A’s `ods_chg_log` table. The filter used for the query is:

   ```
   (& (objectclass=changelogentry)
      (changeNumber>=Last_Transported_ChgNum_From_A_TO_B)
      (! (modifiersname=ReplicaID_of_B))
      (! (orclreplicationid = Orclreplicationid_Attribute_Value_of_Processing_Agreement))
      (!([targetdn=*,excluded_naming_ctx_dn } (..) (..) ..))
      (targetdn=cn=catalogs) (targetdn=cn=subschemasubentry)
      (targetdn=cn=oracleschemaversion) (targetdn=*,included_naming_ctx_dn)
      (targetdn=...) )
   ```

   A control is also passed along with this search operation with the value of `orclreplicationid` specified in the processing agreement.

4. The replication Server at B requests that the Oracle Internet Directory at Replica B store the new change log as a shadow change log.

   4’. The Oracle Internet Directory server at Replica B stores the shadow change log in the `asr_chg_log` table at Replica B.

Steps 3, 3’, and 4 correspond with the transport part of LDAP-based change log processing.
5. The replication server at Replica B queries for new change logs from Replica A from the local change log store asr_chg_log. The filter for the query is:

\[
\langle (\text{objectclass=changeLogEntry}) (\text{servername=ReplicaID_of_A}) \\
\quad (\text{changeNumber}>= \text{Last_Applied_ChgNum_From_A_TO_B})\rangle
\]

6. The replication server at Replica B request that the Oracle Internet Directory server at Replica B apply the retrieved change to storage.

7. Oracle Internet Directory server at Replica B accepts and commits the change to Oracle Internet Directory database storage.

7’ If the local replica is the source of any other replicas, a change log is regenerated in ods_chg_log table at Replica B, which follows the following changelog regeneration rules:

\[
\text{server} = \text{server of local replica, ReplicaB} \\
\text{orclreplicationid/chg_rid} = \text{N} \\
\text{Modifiersname} = \text{Repibind_DN_of_A}
\]

\text{orclreplicationid/chg_rid} is set to the orclreplicationid value of the processing agreement.

8. The replication server requests that the server update the shadow change log retry_cnt status properly.

**Conflict Resolution in Oracle Replication**

Oracle Database Advanced Replication and two-way LDAP-based replication enable updates to multiple directory servers. Conflicts occur whenever the directory replication server attempts to apply remote changes from a supplier to a consumer and, for some reason, fails.

There are times when the replication process may not be able to apply a change. For example, suppose that Supplier Node A sends the consumer a change, and, immediately after that, Supplier Node B sends the consumer an update to the same entry. Then, suppose that a problem delays the transmission of the entry from Supplier Node A, but that no such problem delays transmission of the update from Supplier Node B. The result can be that the update from Supplier Node B arrives at the consumer ahead of the entry it is modifying. In this case, the replication server makes a specified number of retries to apply the change. If it fails to apply the change once that number is reached, then it moves the change to the human intervention queue, and attempts to apply the change at regular, less frequent intervals that you specify.

LDAP operations that can lead to conflicts include:

- Addition
- Deletion
- Modification
- Modification of either an RDN or a DN

**Levels at Which Replication Conflicts Occur**

There are two types of conflicts:

- Entry-level conflicts
- Attribute-level conflicts
Typical Causes of Conflicts

Conflicts usually stem from differences in the timing of changes arising from the occasional slowness or transmission failure over wide area networks. Also, an earlier inconsistency might continue to cause conflicts if it is not resolved in a timely manner.

Automated Resolution of Conflicts

The directory replication server attempts to resolve all conflicts that it encounters by following this process:

1. The conflict is detected when a change is applied.
2. The replication process attempts to reapply the change a specific number of times or repetitively for a specific amount of time after a specific waiting period.
3. If the replication process reaches the retry limit without successfully applying the change, it flags the change as a conflict, which it then tries to resolve. If the conflict cannot be resolved according to the resolution rules (described in the next section), the change is moved to a low-priority, human intervention queue. Changes are then applied according to the time unit specified in the orclHQSchedule parameter in the replication agreement. Before it moves the change, the directory replication server writes the conflict into a log file for the system administrator.

Table 29–11  Types of Replication Conflict

<table>
<thead>
<tr>
<th>Level of Replication Conflict</th>
<th>Description</th>
</tr>
</thead>
</table>
| Entry-level conflicts         | Caused when the directory replication server attempts to apply a change to the consumer. One of the following types of changes to the consumer could occur:
  ■ Adding an entry that already exists
  ■ Deleting an entry that does not exist
  ■ Modifying an entry that does not exist
  ■ Applying a modifyrdn operation when the DN does not exist
  These conflicts can be difficult to resolve. For instance, it may be impossible to resolve a conflict because:
  ■ The entry has been moved to a different location
  ■ The entry has not yet arrived from a supplier
  ■ The entry has been deleted
  ■ The entry never existed on the consumer
  If an entry exists and it should not, then it may be because it was added earlier, or that it recently underwent a modifydn operation. |
| Attribute-level conflicts     | Caused when two directories are updating the same attribute with different values at different times. If the attribute is single-valued, then the replication process resolves the conflict by examining the timestamps of the changes involved in the conflict. |
Replication Failover

As of 10g (10.1.4.0.1), Oracle Internet Directory supports failover of LDAP replicas from one supplier to another. Administrator intervention is required. Figure 29–11 shows a typical failover scenario.

Note: There is no conflict resolution of schema, catalog, and group entries during replication. This is because attempting resolution of such large multivalued attributes would have a significant negative impact on performance. Be careful to avoid updating such entries from more than one master at a time.

See Also:

- Appendix F, "The Multimaster Replication Process" for descriptions of how the multimaster replication process adds, deletes, and modifies entries, and how it modifies DNs and RDNs.
- "Oracle Identity Management LDAP Schema Reference" in Oracle Identity Management User Reference for schema questions
- The catalog command-line tool reference in Oracle Identity Management User Reference for catalog questions
- The section on managing group entries in Oracle Identity Management Guide to Delegated Administration for group entry questions

Replication Failover
Figure 29-11 Replication Failover Scenario

Before Fail-Over

Multimaster Replication DRG

Server Down

Banana

Candy

After Fail-Over

Multimaster Replication DRG

Server Down

Banana

Candy

Oracle Advanced Database Replication
Two-Way LDAP
One-Way LDAP
This scenario has the following features:

- Apple, Banana and Candy are multimaster replicas in the same DRG.
- Damson is a read-only fan-out replica of Banana. That is, it is a partial replica using one-way LDAP replication.
- Eggplant is an updateable fan-out replica of Banana. That is, it is a partial replica using two-way LDAP replication.
- If Banana goes down, replication between the multimaster DRG and its fan-out replicas is broken.

An administrator can switch Eggplant and Damson to a new supplier, Candy.

Only two failover topology types are supported:

- The consumer is connected to the old supplier with an LDAP-based agreement and the old supplier is in the same Advanced Replication group as the new supplier. This is shown in Figure 29–12. Node 2 and Node 3 are in the same Advanced Replication DRG. Node 2 is the original supplier for Node 4. When Node 4 fails, you can fail over Node 4 to a new supplier, Node 3.

  **Figure 29–12 Old and New Suppliers in the Same Advanced Replication Group**

- The consumer and new supplier are both connected to the old supplier with LDAP-based replication agreements. This is shown in Figure 29–13. Node 1 and Node 3 both have LDAP replication agreements with Node 2. Node 2 is the original supplier for Node 1. When Node 2 fails, you can fail over Node 1 to a new supplier, Node 3.

  **Figure 29–13 Consumer and New Supplier Connected to Old Supplier by LDAP**
Included and Excluded Naming Contexts in Partial Replication

In Advanced Replication, you can only exclude naming contexts.

In LDAP-based replication, you can include a given naming context for replication and exclude one or more of the subtrees within that naming context from replication. You can also exclude from replication one or more of the attributes in that naming context.

In LDAP-based replication, only naming contexts explicitly specified as included are replicated. In Oracle Database Advanced Replication, however, all naming contexts are included by default.

To exclude a naming context from replication in Oracle Advanced Replication, specify it using the `orclexcludednamingcontext` attribute of the `Oracle-Advanced-Replication-based-replication agreement entry orclagreementid=000001`.

Figure 29–14 and the accompanying text further exemplify the use of the naming context container and its objects.

Figure 29–14  Example of a Naming Context Container and Its Objects

In Figure 29–14, the naming context included for replication is `c=us`. Within that naming context, one subtree, namely `cn=user1,cn=hr,c=us` is excluded from replication. Moreover, two of the attributes of the `c=us` naming context are excluded from replication—namely, `userPassword` and `telephonenumber`.

See Also: "The Replication Naming Context Object Entry" on page 29-13

Oracle Database Advanced Replication Filtering

This section describes the rules for Oracle Database Advanced Replication filtering.

The following naming contexts cannot be replicated:

- DSE root-specific entry
- `orclagreementid=000001,cn=replication configuration`
- `cn=subconfigsubentry`
- `cn=Oracle Internet Directory`
- `cn=subregistrysubentry`

The following naming contexts cannot be excluded from replication:

- `cn=catalogs`
- `cn=subschemasubentry`
LDAP Replication Filtering

This section describes rules and best practices to follow when specifying naming contexts in LDAP partial replication. It contains the following topics:

- Rules for LDAP Replication Filtering
- Examples of LDAP Replication Filtering
- Rules for Managing Naming Contexts and Attributes
- Optimization of Partial Replication Naming Context for Better Performance

Rules for LDAP Replication Filtering

When two or more naming context objects are configured for replication, the filtering rules are as follows:

1. The overall included naming context is the union of all included naming contexts defined in each naming context object.
2. The overall excluded naming contexts is the union of all excluded naming contexts defined in each naming context object.
3. The attribute exclusions in a naming context object are specific only to that naming context object.
4. If there is a conflict between an included naming context and an excluded naming context, the excluded naming context overrules the included naming context. For example, if an included naming context in naming context object A is a subtree of an excluded naming context specified in another naming context object, B, the subtrees specified in orclecludednamingcontexts of naming context object B will not be replicated. That is, replication filtering in naming context object A will be ignored.

Examples of LDAP Replication Filtering

The discussion in this section relies on the sample naming context illustrated in Figure 29–15. A partial list of user attributes is shown under cn=user1, cn=user2, and cn=user1000.

Figure 29–15  A Sample Naming Context

- cn=oracleschemaversion
- cn=replication configuration

Figure 29–15  A Sample Naming Context
The following examples show how these rules work:

- **Scenario A:** The Included Naming Context in One Naming Context Object Is a Subtree of the Included Naming Context in Another Naming Context Object
- **Scenario B:** The Included Naming Context in One Naming Context Object Is a Subtree of An Excluded Naming Context in Another Naming Context Object

### Rules for Managing Naming Contexts and Attributes

### Optimization of Partial Replication Naming Context for Better Performance

#### Scenario A: The Included Naming Context in One Naming Context Object Is a Subtree of the Included Naming Context in Another Naming Context Object

In this scenario, the included naming context in naming context object #2 is a subtree of the included naming context in object #1.

**Naming Context Object #1**

dn: cn=namectx001,
cn=replication namecontext,
orclagreementid=unique_identifier_of_the_replication_agreement,
orclreplicaid=unique_identifier_of_the_supplier,
cn=replication configuration
orclincludednamingcontexts: cn=mycompany

Naming context object #1 includes the entire DIT under `cn=myCompany`, as shown in Figure 29–16.

**Figure 29–16 Naming Context Object #1**

- **Naming Context Object #2**

dn: cn=namectx002,
cn=replication namecontext,
orclagreementid=unique_identifier_of_the_replication_agreement,
orclreplicaid=unique_identifier_of_the_supplier,
cn=replication configuration
orclincludednamingcontexts: cn=hr,c=us,cn=mycompany
orclexcludednamingcontexts: cn=user1,cn=hr,c=us,cn=mycompany
oracleexcludedattributes: userPassword

Naming context object #2 includes the DIT under cn=hr, c=us, cn=mycompany, but excludes cn=user1 and the attribute userPassword, as shown in Figure 29–17.

**Figure 29–17 Naming Context Object #2**

The result of combining naming context objects #1 and #2 is shown in Figure 29–18.

**Figure 29–18 Result of Combining Naming Context Objects #1 and #2**

In this scenario, the naming context that is replicated is the highest one specified in the orclincludednamingcontexts attribute. Any excluded naming contexts are not replicated. All changes under the subtree cn=mycompany are replicated, except for cn=user1, cn=hr, c=us, cn=mycompany and the attribute userPassword under cn=hr, c=us, cn=mycompany, which are excluded. The attribute userPassword under the rest of the DIT, however, is not excluded from replication because exclusion of userPassword was specified only for naming context object #2, which only included the DIT under cn=hr.

**Scenario B: The Included Naming Context in One Naming Context Object Is a Subtree of An Excluded Naming Context in Another Naming Context Object**

In this scenario, the excluded naming context in naming context object #4 is a subtree of the excluded naming context defined in naming context object #3.
**Naming Context Object #3**

dn: cn=namectx001, cn=replication namecontext,
   orclagreementid=identifier, orclreplicaid=supplier, cn=replication configuration
orclincludednamingcontexts: cn=mycompany
orclexcludednamingcontexts: c=us, cn=mycompany

Naming context object #3 excludes everything under c=us, cn=mycompany, as shown in Figure 29–19.

**Figure 29–19 Naming Context Object #3**

**Naming Context Object #4**

dn: cn=namectx002, cn=replication namecontext, orclagreementid=identifier, orclreplicaid=supplier,
   cn=replication configuration
orclincludednamingcontexts: cn=hr, c=us, cn=mycompany
orclexcludednamingcontexts: cn=user1, cn=hr, c=us, cn=mycompany
orclexcludedattributes: userPassword

Naming context object #4 includes the DIT under cn=hr, c=us, cn=mycompany but excludes user1, as well as the userPassword attribute for all users, as shown in Figure 29–20.

**Figure 29–20 Naming Context Object #4**

The result of combining naming context objects #3 and #4 is shown in Figure 29–21.
In this scenario, the included naming context specified in naming context object #4 is not replicated. That naming context is a subtree of a specified excluded naming context in naming context object #3. In this case, naming context object #4 is ignored, and no changes under `cn=hr,c=us,cn=mycompany` are replicated.

**Rules for Managing Naming Contexts and Attributes**

The following naming contexts cannot be replicated:

- DSE root-specific entry
- `orclagreementid=000001,cn=replication configuration`
- `cn=subconfigsubentry`
- `cn=Oracle Internet Directory`
- `cn=subregistrysubentry`

The following naming contexts cannot be excluded from replication:

- `cn=catalogs`
- `cn=subschemasubentry`
- `cn=oracleschemaversion`
- `cn=replication configuration`

The following attributes cannot be excluded from replication whether they are mandatory or optional. Even if you specify attributes in this list for exclusion from replication, they will always be replicated.

- `orclguid`
- `creatorsname`
- `createtimestamp`
- `cn`
- `dn`
- `attributetypes`
- `objectclasses`
- `objectclass`
You cannot exclude mandatory attributes from replication. For example, suppose that you have an object class named `my_object_class`, which includes the following attributes: mandatory_attribute_1, optional_attribute_1, and optional_attribute_2. In this case, you cannot exclude from replication mandatory_attribute_1.

If you attempt to exclude from replication an attribute that is required for the LDAP operation to be executed, replication will not occur.

If you are configuring partial replication from specific naming contexts in an Oracle Internet Directory node to fan-out replication nodes, then do not change the names of these naming context entries in the source node.

Partial replication will not replicate changes to the root entry of a naming context made by using `ldapmoddn`.

### Optimization of Partial Replication Naming Context for Better Performance

You must plan partial replication carefully to avoid degrading the performance of the replication process. For best performance, use as few naming context objects as possible. For example, the combined use of naming context objects #5 and #6 fulfills the same requirement as the use of naming context object #7, but using naming context object #7 provides better performance.

This section contains these examples:

- **Naming Context Object #5**
- **Naming Context Object #6**
- **Naming Context Object #7**

**Naming Context Object #5**

cn=namectx001,cn=replication
datacontext,orclagreementid=identifier,orclreplicaid=supplier,cn=replication
configuration
orclincludednamingcontexts: cn=mycompany
orclexcludednamingcontexts: c=europe,cn=mycompany
orclexcludedattributes: userPassword

Naming context object #5 is shown in **Figure 29–22**. It includes the DIT under cn=mycompany, but excludes everything under c=europe. It also excludes the attribute userPassword.
Figure 29–22 Naming Context Object #5

Naming Context Object #6

cn=namctx002,cn=replication
namecontext,orclagreementid=<id>,orclreplicaid=<supplier>,cn=replication
configuration
orclincludednamingcontexts: cn=hr, c=us,cn=mycompany
orclexcludednamingcontexts: cn=user1,cn=hr, c=us,cn=mycompany
orclexcludedattributes: userPassword

Naming context object #6 is shown in Figure 29–23. It includes the DIT under cn=hr, c=us, cn=mycompany but excludes user1 and the attribute userPassword.

Figure 29–23 Naming Context Object #6

If naming context objects #5 and #6 are combined, then all changes under cn=mycompany are replicated, except for c=europe,c=mycompany, cn=user1, cn=hr, c=us, cn=mycompany, and the attribute userPassword.

You could fulfill the same requirement, however, by using naming context object #7. Using a single naming context object provides better partial replication performance.

Naming Context Object #7

cn=namctx001,cn=replication
namecontext,orclagreementid=identifier,orclreplicaid=supplier,cn=replication
configuration
orclincludednamingcontexts: cn=mycompany
orclexcludednamingcontexts: c=europe,cn=mycompany
orclexcludednamingcontexts: cn=user1, cn=hr, c=us, cn=mycompany
orclexcludedattributes: userPassword

Naming context object #7 is shown in Figure 29–24.
Replication is the mechanism that maintains exact duplicates of specified naming contexts on multiple nodes. This chapter tells you how to install and configure replication in Oracle Internet Directory.

At 10g (10.1.4.0.1), there are four types of installation for replication. They are:

- Installing Oracle Internet Directory as a master
- Installing Oracle Internet Directory as an Advanced Replication-based replica
- Installing Oracle Internet Directory as a one-way LDAP-based replica
- Installing Oracle Internet Directory as a two-way LDAP-based replica

The procedures for installing the three types of replica are very similar, so they are grouped into one topic. The few differences are pointed out in the procedures.

This chapter contains these topics:

- Oracle Internet Directory Versions and Replication
- Preliminary Information for Installing and Configuring a Replication Group
- Installing and Configuring Multimaster Replication
- Installing and Configuring One-Way or Two-Way LDAP-Based Replication
- Resolving Conflicts Manually in a Replication Group
- Example: Installing and Configuring a Multimaster Replication Group with Fan-Out
- Configuring Replication Failover

**Oracle Internet Directory Versions and Replication**

For either multimaster replication or two-way fan-out replication, if any node in the replication group is running 10g (10.1.4.0.1), then all nodes must be running 10g (10.1.4.0.1).

In a one-way fan-out deployment, the supplier of a 10g (10.1.4.0.1) consumer can be running 10g Release 3 (10.1.2.0.2). This is true for a supplier of either LDAP-based replication or Advanced Replication-based replication.
Preliminary Information for Installing and Configuring a Replication Group

This section describes the types of installation you need to perform to configure a replication group. It also introduces the Replication Environment Management Tool, which enables you to perform various configuration tasks.

This section contains the following topics:

- Oracle Internet Directory Installation
- If You are Installing Oracle Internet Directory as a Master
- If You are Installing Oracle Internet Directory as an Advanced Replication-Based Replica or as a One-Way or Two-Way LDAP-Based Replica
- The Replication Environment Management Tool

Oracle Internet Directory Installation

When you install Oracle Internet Directory as part of Oracle Application Server on any node, you are prompted to select a product. Choose the Oracle Application Server Infrastructure.

Later in the installation process, you are prompted to choose an installation type.

- For multimaster replication, you need a single master definition site (MDS). For that, follow the directions in the section entitled "If You are Installing Oracle Internet Directory as a Master" on page 30-3.
- Each other site is a replica, either an Advanced Replication-based replica or an LDAP replica.
  - Oracle Database Advanced Replication-based replicas do true multimaster replication: changes made to the master are replicated to the replicas and changes made to any replica are replicated to the master and all other replicas.
  - Although a site that will be used as an Advanced Replication-based replica can be initially installed as a master, Oracle recommends against that option. Oracle recommends installing each intended replica as a replica right from the start.
    
    Therefore, to create an Advanced Replication-based replica, follow the directions in the section entitled "If You are Installing Oracle Internet Directory as an Advanced Replication-Based Replica or as a One-Way or Two-Way LDAP-Based Replica" on page 30-3, and choose "Advanced Replication" when that choice appears.

  - LDAP replicas can be either one-way or two-way.
    
    * One-way LDAP replicas are read-only. Only changes made to the master are replicated to the replica. There is no replication from the replica to the master.
      
      To create a one-way LDAP replica, follow the directions in the section entitled "If You are Installing Oracle Internet Directory as an Advanced Replication-Based Replica or as a One-Way or Two-Way LDAP-Based Replica" on page 30-3, and choose "One-way LDAP replica" when that choice appears.
    
    * Two-way LDAP replicas are updatable in both directions. Changes made to either replica are replicated to the other.
To create a two-way LDAP replica, follow the directions in the section entitled "If You are Installing Oracle Internet Directory as an Advanced Replication-Based Replica or as a One-Way or Two-Way LDAP-Based Replica" on page 30-3, and choose "Two-way LDAP replica" when that choice appears.

LDAP replicas can also be used to replicate a portion of the directory information tree rather than the entire directory information tree. In partial replication, you determine what is or is not replicated by defining replica naming context objects. You can do this for both one-way and two-way replicas.

See Also: Detailed explanations and examples of setting up and using naming contexts appear in later sections of this chapter.

If You are Installing Oracle Internet Directory as a Master

1. Follow the installation procedure documented in the chapter "Installing Oracle Internet Directory in Replicated Mode" in the Oracle Application Server Installation Guide. When the Oracle Universal Installer prompts you to select a product to install, choose the Oracle Application Server Infrastructure.

2. For the Installation Type, select as follows:
   a. If you do not have (or will not be using) an existing Oracle Application Server Metadata Repository, choose Identity Management and Oracle Application Server Metadata Repository.
   b. If you wish to use an existing Oracle Application Server Metadata Repository, choose Identity Management.

3. Ensure that Oracle Internet Directory is checked on the Select Configuration Options screen.

4. When installing a master, do not check High Availability and Replication in the Select Configuration Options screen for replication. When you do not check High Availability and Replication, Oracle Universal Installer will perform a default Oracle Internet Directory install, that is, it will install a new Oracle Internet Directory as a master node. (You may still check High Availability and Replication in the Select Configuration Options screen for configuring Virtual Host or OracleAS Clusters though.)

5. Complete the installation as described in "Installing Oracle Internet Directory in Replicated Mode" in the Oracle Application Server Installation Guide.

If You are Installing Oracle Internet Directory as an Advanced Replication-Based Replica or as a One-Way or Two-Way LDAP-Based Replica

1. Follow the installation procedure documented in the chapter "Installing Oracle Internet Directory in Replicated Mode" in the Oracle Application Server Installation Guide. When the Oracle Universal Installer prompts you to select a product to install, choose the Oracle Application Server Infrastructure.

2. For the Installation Type, select as follows:
   a. If you do not have (or will not be using) an existing Oracle Application Server Metadata Repository, choose Identity Management and Oracle Application Server Metadata Repository.
b. If you wish to use an existing Oracle Application Server Metadata Repository, choose **Identity Management**.

3. For Configuration Options, ensure that both **Oracle Internet Directory** and **High Availability and Replication** are checked.

4. Because you checked **High Availability and Replication** in the Select Configuration Options screen, you will see the Select High Availability or Replication Option screen. Select **Replication**.

5. Next, you will see the **Oracle Internet Directory Replication Mode** screen. Choose the type of replica you want to create:
   - For Advanced Replication-based (multimaster) replication, select **Advanced Replication**.
   - For a one-way (read-only) LDAP replica, select **One-Way LDAP Replica**.
   - For a two-way (read-write) LDAP replica, select **Two-Way LDAP Replica**.

6. On the screen labeled **Oracle Internet Directory Master Node**, specify the hostname and port for the supplier node to be replicated by this node presently being created. If connecting with that node requires SSL protocol, check that box on this screen.

7. Complete the installation as described in Oracle Application Server Installation Guide.

### The Replication Environment Management Tool

During installation and configuration, you use the Replication Environment Management Tool to perform various tasks. This tool assists you in:

- Configuring a replication group
- Adding and deleting replicas
- Managing the directory replication group
- Modifying or resetting the replication Bind DN password
- Modifying the database replication user REPADMIN password
- Displaying various errors and status information for change log propagation

**Note:** You do not need Advanced Replication to perform partial—that is, LDAP-based—replication.

Even in a directory replication group whose nodes have different patchset versions of Oracle Database 10g, you can replicate if they have the same version of Oracle Internet Directory.

However, if the nodes in a directory replication group are running different versions of Oracle Internet Directory, there is a constraint on modifying directory servers on those nodes: Do not replicate changes generated on a newer version of Oracle Internet Directory to a node that has not yet upgraded to that version. If you do, the changes can contain information that the earlier version cannot properly interpret.
See Also: The remtool command-line tool reference in Oracle Identity Management User Reference for more information about the Replication Environment Management Tool

Installing and Configuring Multimaster Replication

This section tells you how to install and configure multimaster replication groups, and how to resolve conflicts manually in them. It contains these topics:

- Rules for Configuring Directory Replication Based on Oracle Database Advanced Replication
- Installing and Configuring a Multimaster Replication Group
- Adding a Node for Multimaster Replication (Oracle Database Advanced Replication Types Only)
- Deleting a Node from a Multimaster Replication Group

See Also:

- "Installing Oracle Internet Directory in Replicated Mode" in Oracle Application Server Installation Guide
- The chapters on Multimaster Replication in Oracle Application Server High Availability Guide

Rules for Configuring Directory Replication Based on Oracle Database Advanced Replication

The following nine rules apply to replication based on Advanced Replication (sometimes referred to as ASR):

1. In this type of Directory Replication Group (DRG), there must be one node identified as the Master Definition Site (MDS): this is the group master. All other nodes taking part in the replication are replicas, which in database replication are termed "Remote Master Sites" (RMS). The MDS must be created as a master node, not as a replica, that is, neither as an Advanced Replication-based replica nor as an LDAP-based replica.

Note: Even though it is not the central master, an Advanced Replication-based replica is sometimes called a remote master site (RMS), due to two facts. The first is that in Advanced Replication, when information is moved from one site to another, the recipient of the transferred information is called a "remote master site." The second fact is that independent changes made directly to an Advanced Replication-based replica are also replicated to all members of its group, making it a "master" during that interaction. Such a group, in which changes to any member are replicated to all other members, is called a multimaster replication group.
Installing and Configuring Multimaster Replication

2. When you install and configure Multimaster replication, the master node for a Directory Replication Group (DRG) and each node that is to become an Advanced Replication-based replica must be initially empty, that is, a new Oracle Internet Directory installation.

   **Note:** If the Master node is not a new installation, use the procedure described in "Adding a Node for Multimaster Replication (Oracle Database Advanced Replication Types Only)" on page 30-14 to add replicas. That procedure will also initialize the replication group.

3. When you add an Oracle Database Advanced Replication-based replica, the new replica must be empty. That is, Oracle Internet Directory must be newly installed.

4. The sponsor node for each Advanced Replication-based replica can be any of the following:
   - A master node
   - An Advanced Replication-based replica of an existing multi-master DRG
   - A supplier of an LDAP replica that is not a consumer LDAP replica of any other LDAP replica

5. An Advanced Replication-based replica cannot be a consumer of an LDAP replica.

6. In Oracle Internet Directory 10g (10.1.4.0.1), a node cannot be part of more than one multimaster replication group.

7. The data replicated between servers in a directory replication group does not include DSE root-specific data, server configuration data, and replication agreement data.

   **See Also:** "Oracle Database Advanced Replication Filtering" on page 29-30

8. When an multimaster replication group is configured, the Oracle Application Server Single Sign-On database schema is automatically configured in replication.

9. When you add a node to a DRG, it must be running the same version of Oracle Internet Directory as the other nodes in the DRG. If you want to add a new 10g (10.1.4.0.1) node to a DRG containing nodes at an earlier release, first upgrade all existing nodes to 10g (10.1.4.0.1).

---

**Installing and Configuring a Multimaster Replication Group**

This section discusses the general tasks you perform when installing and configuring a multimaster replication group. It contains these topics:
Task 1: Install Oracle Internet Directory as a Master on the Master Definition Site (MDS) on page 30-7

Task 2: Install the Oracle Internet Directory as a Replica, on the Remote Master Sites (RMS) on page 30-7

Task 3: Set Up Oracle Database Advanced Replication for a Directory Replication Group on page 30-9

Task 4 (Optional): Load Data into the Directory on page 30-12

Task 5: Ensure that Oracle Directory Server Instances are Started on All the Nodes on page 30-13

Task 6: Start the Replication Servers on All Nodes in the DRG on page 30-13

Task 7: Test Directory Replication on page 30-14

Note:

- The instructions in this section apply to setting up replication in a group of empty nodes. They assume that there is no pre-existing directory data on any of the nodes in the DRG. For instructions on adding a node to an existing DRG, see "Adding a Node for Multimaster Replication (Oracle Database Advanced Replication Types Only)" on page 30-14.

- During entry replication, the directory replication server does not always preserve the spaces between RDN components in the DN. In some rare cases, it may not preserve the case of the letters in the DN.

Task 1: Install Oracle Internet Directory as a Master on the Master Definition Site (MDS)

You must be able to use Oracle Net Services to connect to the master definition site database and all other nodes in the DRG.

Note: During installation, make sure that each Oracle Internet Directory database instance name is unique on each machine.

To do the actual installation, as a master, use the instructions in the section entitled "If You are Installing Oracle Internet Directory as a Master" on page 30-3.

See Also: Installing Oracle Internet Directory in Replicated Mode" in Oracle Application Server Installation Guide.

Task 2: Install the Oracle Internet Directory as a Replica, on the Remote Master Sites (RMS)

Oracle recommends that the sites to be used as Advanced Replications-based replicas be installed initially as replicas, and not as masters.

To do the actual installation, as a replica, use the instructions in the section entitled "If You are Installing Oracle Internet Directory as an Advanced Replication-Based Replica or as a One-Way or Two-Way LDAP-Based Replica" on page 30-3.
If an Existing Master is Used as a Remote Master Site Although Oracle recommends starting with empty replicas, you can set up replication using machines initially configured as masters rather than replicas. To use a machine initially configured as a master as an RMS, you must first migrate its metadata to the MDS, as follows:

- Make sure the Oracle Internet Directory server is up and running on both the MDS and each such desired replica so that the process (remtool -backupmetadata) can succeed.

- From the newly created node, run the following command:

  ```
  remtool -backupmetadata \
  -replica "new_node_host:new_node_port/new_node_repldn_pwd" \
  -master "master_host:master_port/master_repl_dn_pwd"
  ```

  where `master_host:master_port/master_repl_dn_pwd` are the hostname, port number, and replication DN password for the desired replica's supplier.

  **Note:** If Oracle Delegated Administration Services is not configured, you might see an error message similar to this when you run remtool with the -backupmetadata option:

  ```
  Failed to add "orclApplicationCommonName=ias.acme.com, cn=IAS Instances, cn=IAS, cn=Products, cn=OracleContext" as "uniquemember" to entry "cn=Associated Mid-tiers, orclapplicationcommonname=DASApp, cn=DAS,cn=products, cn=OracleContext at replica ldap://myhost:389
  ```

  Please ignore this error message.

- Apart from loading the metadata into master replica, this tool creates a file named `ocbkup.new_replica_id.TO.master_replica_id.timestamp.dat` containing the metadata as backup. This file is created under the `$ORACLE_HOME/ldap/log` directory. This file contains the changes made to master replica in LDIF format, a copy of SSO container entry [orclApplicationCommonName=ORASSO_SSOSERVER, cn=S SO, cn=Products, cn=OracleContext] and DAS URL container entry [cn=OperationURLs, cn=DAS, cn=Products, cn=OracleContext].

- If the metadata backup succeeded, it displays a message in the terminal:

  ```
  Backup of metadata will be stored in $ORACLE_HOME/ldap/log/ocbkup.new_replica_id.TO.master_replica_id.timestamp.dat.
  Metadata copied successfully.
  ```

  The message will contain the actual path of `ORACLE_HOME`.

- If an error occurs during this operation, remtool reports the error in the terminal from which it was invoked. The error messages are also logged in `$ORACLE_HOME/ldap/log/remtool.log` file.

After successfully migrating the master's metadata to the MDS, you can now safely continue with "Task 3: Set Up Oracle Database Advanced Replication for a Directory Replication Group" on page 30-9.
Task 3: Set Up Oracle Database Advanced Replication for a Directory Replication Group

The following sections lead you through installing and configuring Advanced Replication by using the Replication Management Tool.

See Also: Oracle Database Advanced Replication in the Oracle Database Documentation Library, and the online Help for the Replication Management Tool, for information on configuring Oracle Database Advanced Replication.

To establish a directory replication group (DRG), you must configure the Advanced Replication environment by performing the tasks discussed in these topics:

- On All Nodes, Prepare the Oracle Net Services Environment for Replication
- From the MDS, Configure Oracle Database Advanced Replication For Directory Replication

On All Nodes, Prepare the Oracle Net Services Environment for Replication For each node in the directory replication group, perform the steps listed here. (Each step is described more fully in the subsections that directly follow this list.)

1. Configure sqlnet.ora.
2. Configure tnsnames.ora.
3. Stop and restart the listener.
4. Test Oracle Net connections to all nodes from each node in the DRG.

To prepare the Oracle Net Services environment for replication:

1. Configure sqlnet.ora.

The sqlnet.ora file should contain the following parameters at minimum:

names.directory_path = (TNSNAMES)
names.default_domain = global_database_domain

On UNIX, the sqlnet.ora file is in $ORACLE_HOME/network/admin.

On Microsoft Windows, the sqlnet.ora file is in %ORACLE_HOME\network\admin.

2. Configure tnsnames.ora.

On each node in the DRG, define all Oracle Internet Directory database instances in the DRG. Each tnsnames.ora file must contain connect descriptor information in the following format for each Oracle Internet Directory database:

net_service_name =
  DESCRIPTION =
  ADDRESS =
  PROTOCOL = TCP
  HOST = HOST_NAME_OR_IP_ADDRESS
  PORT = port_no_of_listener)
  CONNECT_DATA =
    (service_name = service_name_of_database))

where net_service_name is the global name of the database. For example, if the database global name is mds.sales.com, then your net_service_name must be mds.sales.com. Ensure that your database global name and your net_
service_name are domain-qualified. In this example, the global name and net_

---

**Note:**

- The database global name is composed of the DB_NAME and DB_DOMAIN initialization parameters of your database. For example, if your database's DB_NAME is mds and DB_DOMAIN is sales.com, your database global name is mds.sales.com. The global name will not be domain qualified if the DB_DOMAIN initialization parameter is not defined.

- The value of the NAMES.DEFAULT_DOMAIN parameter in the sqlnet.ora file must match the value of the DB_DOMAIN initialization parameter of the database.

---

**See Also:** Oracle Database Net Services Reference for more information on tnsnames.ora syntax.

On UNIX, the tnsnames.ora file is in $ORACLE_HOME/network/admin.

On Microsoft Windows, the tnsnames.ora file is in %ORACLE_HOME%\network\admin.

---

**Note:** You must domain-qualify the net service name (for example, sales.com), but be sure that the domain component matches the one specified in the NAMES.DEFAULT_DOMAIN parameter in the sqlnet.ora file.

---

3. Stop and restart the listener.

To stop the listener for the Oracle Internet Directory database, use the listener control utility (lsnrctl). Type the following command at the LSNRCTL command prompt:

```
SET PASSWORD password
STOP [listener_name]
```

**SET PASSWORD** is required only if the password is set in the listener.ora file. The password defaults to ORACLE. The default listener name is LISTENER.

To restart the listener for the Oracle Internet Directory database, type the following command at the LSNRCTL command prompt:

```
START [listener_name]
```

quit

4. Test Oracle Net connections to all nodes from each node in the DRG.

**IMPORTANT:** Try to connect using both of these commands:

```
sqlplus ods/ods_password@net_service_name_without_domain_name
sqlplus ods/ods_password@net_service_name_with_domain_name
```

If you cannot connect, then replication will not work.
Installing and Configuring Multimaster Replication

From the MDS, Configure Oracle Database Advanced Replication For Directory Replication

To do this:

1. From the MDS console, connect as the system user on all nodes, including the MDS. Ensure the following on all nodes:
   - The Oracle Internet Directory database is running
   - The Oracle Internet Directory listener is running
   - The connect string is correct
   - The system password is correct

2. Ensure the following wallets exist on the remote sites:
   - A wallet for storing the password to the database designated for Oracle Internet Directory. This wallet is named `oidpwdldap1` and is located in the directory `$ORACLE_HOME/ldap/admin`.
   - A wallet for storing the password of the replication administrator. This wallet is named `oidpwdoracle_sid`, and is located in the directory `$ORACLE_HOME/ldap/admin`. (The `oracle_sid` is obtained not from the environment variable SID but from the connected database.)

   If the wallets do not exist on a specific site, create them by typing the following command on the remote node:
   ```
   oidpasswd connect=connect_string create_wallet=true
   ```

3. Check the prerequisites in the attached Note. Then, at a command prompt in the MDS, use remtool (the Replication Environment Management Tool) to configure Advanced Replication by running the following script:
   ```
   $ORACLE_HOME/ldap/bin/remtool -asrsetup
   ```

**Note:**

- The `remtool` command-line tool reference in *Oracle Identity Management User Reference* for information about using the `-asrsetup` option of the Replication Environment Management Tool (`remtool`).
- *Oracle Database Administrator’s Guide* in the Oracle Database Documentation Library for instructions on ensuring that the database and listener are running
- *Oracle Database Net Services Administrator’s Guide* in the Oracle Database Documentation Library for instructions on ensuring that the connect string is correct

**Note:** If you encounter errors, then clean up the environment by using the `-asrcleanup` option of the Replication Environment Management Tool. Then repeat Step 3.
Task 4 (Optional): Load Data into the Directory

You can choose either of two ways to load data into the directory:

- To add just a small number of entries to the DRG, you can wait until you have completely configured the DRG. Then use `ldapadd` to load the data to one of the nodes. The entries will then be replicated to the other nodes at the specified time.

- To add a large amount of data to load into the DRG, use the `bulkload` utility:
  a. Stop the LDAP server at all nodes of the DRG by typing:
     ```
     opmnctl stopproc ias-component=OID
     ```
  b. On the node that is part of the DRG and where you have the ldif file to be loaded onto the directories enter:
     ```
     bulkload connect="connect_string" check="TRUE" \ 
     generate="TRUE" file="file_with_absolute_path_name"
     ```

  Note: If data is extracted from Oracle Internet Directory using `ldifwrite`, then, in addition to other options, use the `restore="TRUE"` option to restore the operational attributes.

  c. On the same node, enter:
     ```
     bulkload connect="connect_string_1" load="TRUE"
     ```

     Repeat step c on the same node, each time replacing `connect_string_1` with the connect string of another node in the DRG, until you have loaded the data onto all the nodes in the DRG. For example, enter:
     ```
     bulkload connect="connect_string_2" load="TRUE"
     ```

     then enter
     ```
     bulkload connect="connect_string_3" load="TRUE"
     ```

     and so on, until you loaded the data onto all the nodes in the DRG.

  Note:
  - `connect_string` is the connect string of the local Oracle Internet Directory database.
  - For successful replication, an entry must have the same `orclguid` (global identifier) at all replicated nodes. This is accomplished by performing Step b once and repeating Step c for all nodes in the DRG.
**Task 5: Ensure that Oracle Directory Server Instances are Started on All the Nodes**

The out-of-box configuration has Oracle Internet Directory LDAP Server instance #1 configured with change logging set to TRUE. This default instance of Oracle Internet Directory LDAP Server can be started using `opmn` as follows:

```
opmnctl startproc ias-component=OID
```

**Note:** Do not use the `opmnctl startall` command at this point. If you do, the HTTP server and the Oracle Internet Directory server will start, but the command will hang while trying to start the OC4J server. The OC4J server can be started only after Task 6 is complete. Once the replication server has been started on all nodes of the DRG and replication has propagated to all nodes, you may safely use the `opmnctl startall` command.

**See Also:** The `bulkload` command-line tool reference in *Oracle Identity Management User Reference* for syntax and usage notes

**Task 6: Start the Replication Servers on All Nodes in the DRG**

To start replication servers on all nodes, type the following command on each node:

```
oidctl connect=connect_string server=oidrepld instance=1 \
flags="-h host_on_which_the_directory_server_is_running -p port" start
```

Note that the instance number need not be unique across the entire DRG.

**Note:** If you are deploying a single master with read-only replica consumers, you can reduce performance overhead by turning off the multimaster flag in the directory replication server. To do so, change the value of the `-m` flag in the OID Control Utility command for Oracle directory replication server from the default (TRUE) to FALSE. The multimaster option controls conflict resolution, which serves no purpose if you are deploying a single master.

Once the Oracle Internet Directory replication server has been started using `oidctl`, any `opmnctl` command to stop or start the Oracle Internet Directory component will ensure that the Oracle Internet Directory replication server process is also stopped or started.

**See Also:**

Chapter 6, "Process Control of Oracle Internet Directory Components" for information on Oracle Internet Directory process control.

"Conflict Resolution in Oracle Replication" on page 29-25

Chapter 7, "Oracle Directory Server Administration" for information on starting the replication servers
Task 7: Test Directory Replication

Use Oracle Directory Manager to verify that the directory replication servers are running, and then test directory replication by doing the following:

1. Log in to Oracle Directory Manager as orcladmin.
2. In the navigator pane, expand in succession Oracle Internet Directory Servers, directory server instance, Entry Management.
3. Create a single entry on the MDS node.

The identical entry appears in approximately 1 to 10 minutes on the RMS. You can adjust the timing in the replication server configuration set entry. If entries are modified on any nodes in the DRG, then the changes will be replicated.

Note: If you want to configure replication for Oracle Application Server Single Sign-On, then follow the post-installation steps specific to Oracle Application Server Single Sign-On. These are found in the replication installation section of the Oracle Application Server Single Sign-On Administrator’s Guide.

Adding a Node for Multimaster Replication (Oracle Database Advanced Replication Types Only)

Note: A new node that you add to an existing multimaster replication group must have an Oracle Application Server Infrastructure installed on it. During its installation, the installation type selected had to have been Oracle Application Server with Metadata Repository. For more information, see “Task 2: Install the Oracle Internet Directory as a Replica, on the Remote Master Sites (RMS)” on page 30-7.

You can add a node to a master node, or to an LDAP-based supplier replica that is not a consumer of any other LDAP based replicas, to form a multimaster DRG. When you do so, the steps in this section will automatically perform an initial install and configuration of Advanced Replication.

To add a new replication node to a live, functioning replication group or to a master node of any significant size, perform the following steps:

- Prepare the Oracle Net Services Environment
- Task 1: Stop the Directory Replication Server on All Nodes
- Task 2: Identify a Sponsor Node and Install Oracle Internet Directory as a Replica on the Remote Site
- Task 3: Switch the Sponsor Node to Read-Only Mode
- Task 4: Back up the Sponsor Node by Using ldifwrite
- Task 5: Perform Advanced Replication Add Node Setup
- Task 6: Switch the Sponsor Node to Updatable Mode
- Task 7: Start the Directory Replication Server on All Nodes Except the New Node
- Task 8: Load Data into the New Node by Using bulkload
Task 9: Start the Directory Server on the New Node

Task 10: Start the Directory Replication Server on the New Node

Note: Commands shown in the following tasks require the following types of items to be stored as follows:
- Binaries: $ORACLE_HOME/bin
- SQL scripts: $ORACLE_HOME/ldap/admin
- UNIX scripts: $ORACLE_HOME/ldap/bin

Prepare the Oracle Net Services Environment
The process that prepares this environment is described in Section , "On All Nodes, Prepare the Oracle Net Services Environment for Replication".

Task 1: Stop the Directory Replication Server on All Nodes
To stop the directory replication server, run the following command on each node in the LDAP replication group, type:

oidctl connect=db_connect_string server=oidrepld instance=1 stop

Note: The instance number might not be 1. Check the running process to discover the instance number in use here.

Task 2: Identify a Sponsor Node and Install Oracle Internet Directory as a Replica on the Remote Site
You must identify a sponsor node for this Task. It is the node that will supply the data to the new node.

For the RMS, Oracle recommends that you install the new instance of Oracle Internet Directory as an Advanced Replication replica. (You could use an existing master node as the RMS, but extra manual steps are required.)

To install a new Oracle Internet Directory as an Advanced Replication replica for RMS, use the instructions in "If You are Installing Oracle Internet Directory as an Advanced Replication-Based Replica or as a One-Way or Two-Way LDAP-Based Replica" on page 30-3 and choose Advanced Replication when that choice appears.

If an existing master is used as RMS, you need to follow the instructions in "If an Existing Master is Used as a Remote Master Site" on page 30-8 to migrate the master's metadata to the sponsor node. After successfully migrating the master's metadata to the MDS, you can now safely continue with "Task 3: Switch the Sponsor Node to Read-Only Mode".

Task 3: Switch the Sponsor Node to Read-Only Mode
A sponsor node is the node that supplies the data to the new node. To switch the sponsor node to read-only mode:

1. Create a new file, change_mode.ldif, containing the following:

   dn:
changetype: modify
replace: orclservermode
orclservermode: r

2. Run the following command against the identified sponsor node:

```
ldapmodify -D "cn=orcladmin" -w adminPassword -h host_name_of_sponsor_node \ 
-p port -f change_mode.ldif
```

This command changes the Oracle directory server `name_of_sponsor_node` from sponsor mode to read-only mode.

---

**Note:** While the sponsor node is in read-only mode, you may not make any updates to it. You may, however, update any of the other nodes, but those updates are not replicated immediately.

Also, the sponsor node and the MDS may be the same node.

---

**Task 4: Back up the Sponsor Node by Using ldifwrite**

Because this may take a long time, you may start "Task 5: Perform Advanced Replication Add Node Setup" while backup is in progress.

On the sponsor node, enter the following command:

```
ldifwrite connect="connect_string" \ 
    basedn="orclAgreementID=000001,cn=replication configuration" \ 
    file="output.ldif_file"
```

This backs up the directory of the sponsor node.

---

**Task 5: Perform Advanced Replication Add Node Setup**

---

**Note:** Oracle Net Service must be configured properly on all nodes for replication. See: "On All Nodes, Prepare the Oracle Net Services Environment for Replication" on page 30-9.

---

You can perform the Advanced Replication add node setup at the same time that you perform "Task 4: Back up the Sponsor Node by Using ldifwrite" on page 30-16.

On the sponsor node, enter this command:

```
remtool -addnode
```

The Replication Environment Management Tool adds the node to the DRG.
Task 6: Switch the Sponsor Node to Updatable Mode

To switch the sponsor node to updatable mode:

1. Edit `change_mode.ldif` to the following:

   ```
   dn:
   changetype: modify
   replace: orclservermode
   orclservermode: rw
   ```

2. Run the following commands on the sponsor node:

   ```
   ldapmodify -D adminDN -w adminPassword -h host_name_of_sponsor_node -p port -f change_mode.ldif
   ```

   This command changes the Oracle directory server `host_name_of_sponsor_node` from sponsor mode back to read/write mode.

   **Note:** Task 6 is very similar to Task 3. The only difference is that the `orclservermode` parameter in `change_mode.ldif` is being set back to `rw`, that is, read/write, in this step.

Task 7: Start the Directory Replication Server on All Nodes Except the New Node

To start the directory replication server, type the following command on all nodes except the new node:

```
oidctl connect=db_connection_string server=oidrepld instance=1 \
flags="-h host -p port" start
```

To ensure that no directory or replication processes are running on the new node, type:

```
opmnctl stopproc ias-component=OID
```
Task 8: Load Data into the New Node by Using bulkload

To load data, type the following command on the new node:

```
bulkload connect="db_connect_string_of_new_node" check="TRUE" generate="TRUE" load="TRUE" restore="TRUE" file="absolute_path_to_the_ldif_file_generated_by_ldifwrite"
```

**Note:** If you load data from an earlier version of Oracle Internet Directory, such as 10g Release 2 (10.1.2.0.2) onto a node running 10g (10.1.4.0.1), you must update the password policy entries as described in "Password Policy and Fan-out Replication" on page 30-32.

Task 9: Start the Directory Server on the New Node

To start the directory server, type the following command on the new node:

```
opmctl startproc ias-component=OID
```

Task 10: Start the Directory Replication Server on the New Node

**Note:** If you need to change configuration or agreement parameters, see Chapter 31, "Oracle Internet Directory Replication Monitoring and Management".

To start the directory replication server, type the following command on the new node:

```
oiddctl connect=db_connect_string_of_new_node server=oidrepld instance=1 flags="-p port" start
```

**Note:** Once a directory server instance is participating in a replication agreement, do not use the bulkload tool to add data into the node. Instead, use ldapadd.

If Oracle Application Server Single Sign-On is desired in replication, then follow the Oracle Application Server Single Sign-On Administrator’s Guide in the replication installation section for the post-installation steps specific to Oracle Application Server Single Sign-On.

Deleting a Node from a Multimaster Replication Group

At times, you may want to delete a node from a DRG (for example, if the addition of a new node did not fully succeed as a result of system errors).

To delete a replication node, perform the tasks described in these topics:

- **Task 1:** Stop the Directory Replication Server on All Nodes
- **Task 2:** Stop All Oracle Internet Directory Processes in the Node to be Deleted
- **Task 3:** Delete the Node from the Master Definition Site
- **Task 4:** Start the Directory Replication Server on All Nodes
Task 1: Stop the Directory Replication Server on All Nodes
To stop the directory replication server, run the following command on each node in the DRG:

oidctl connect=connect_string server=oidrepld instance=1 stop

Note: The instance number may vary.

Task 2: Stop All Oracle Internet Directory Processes in the Node to be Deleted
On the node to be deleted, shut down Oracle Internet Directory using opmn.

$ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=OID

See Also: The opmn command-line tool reference in Oracle Identity Management User Reference for more information about shutting down Oracle Internet Directory.

Task 3: Delete the Node from the Master Definition Site
From the MDS, run the following script:

remtool -delnode

The Replication Environment Management Tool deletes the node from the replication group.

See Also: The remtool command-line tool reference in Oracle Identity Management User Reference for instructions on using the -delnode option of the Replication Environment Management Tool

This process can take a long time, depending on your system resources and the size of your DRG. If you use the -v option, the tool keeps you informed of its progress.

Note: If you encounter errors, then use the -asrverify option first. If it reports errors, then rectify them by using the -asrrectify option. Both -asrverify and -asrrectify list all nodes in the DRG. If the node to be deleted is in the list, then delete it by running the Replication Environment Management tool again, using the -delnode option.

Task 4: Start the Directory Replication Server on All Nodes
To start the directory replication server, type the following command on each of the remaining nodes of the DRG:

oidctl connect=connect_string server=oidrepld instance=1 start

flags='-h host -p port' start

See Also: The opmn command-line tool reference in Oracle Identity Management User Reference
Rules for Configuring LDAP-Based Replication

The following rules apply to both full and partial LDAP-based replication:

1. In LDAP-based replication, only the naming contexts listed in the `namingcontexts` attribute of the root DSE can be replicated to the consumer.

   See Also: The discussion of namingcontexts in:
   - "Replication Fields in Oracle Directory Manager" on page A-12
   - "Included and Excluded Naming Contexts in Partial Replication" on page 29-30
   - “Viewing and Modifying Replica Naming Context Objects by Using Oracle Directory Manager” on page 30-34

2. The supplier of an LDAP-based replica can be a master node that is a standalone node, a member of a multimaster replication group, or another LDAP-based replica.

   See Also: For instructions on installing on a standalone node, see "If You are Installing Oracle Internet Directory as a Master" on page 30-3

3. An LDAP-based replica can be a consumer for another LDAP-based replica. That consumer is then called a fan-out replica.

4. You can add an Oracle Internet Directory 10g (10.1.4.0.1) LDAP replica to an Oracle Internet Directory 10g master that is running an earlier version, such as 10g Release 2 (9.0.4) or 10g Release 2 (10.1.2). You can also upgrade an LDAP replica of a 10g (9.0.4) or (10.1.2 master to 10g (10.1.4.0.1). Be aware, however, that earlier releases do not support all the features of 10g (10.1.4.0.1).

   Note: Make sure the schemas are synchronized. Otherwise, the replication server might not be able to apply changes to the consumer replica.

5. Two-way LDAP-based replication is not backward compatible. It is only supported between replicas that are both running 10g (10.1.4.0.1).

6. The new consumer node must be empty. That is, Oracle Internet Directory must be newly installed.
Back Up Your LDAP Data by Using ldifwrite and bulkload

Use the ldifwrite utility to back up LDAP data with operational attributes preserved. Once this is done, use the bulkload utility to load data to all replicas in a group.

Use bulkload with the check="TRUE", generate="TRUE", and restore="TRUE" arguments, and then with the load="TRUE" argument. Preserve the operational attributes by using the same intermediate files (generated by using the generate="TRUE" argument) for all replicas. As of 10g (10.1.4.0.1), you can load multiple replicas in the same invocation of the bulkload command by using connect="connect_string" with the appropriate connect string for each replica.

Backup using this method can take a long time for a directory with one million entries.

See Also:

- Chapter 9, "Using Bulk Tools"
- "Oracle Identity Management Command-line Tool Reference" in Oracle Identity Management User Reference

Installing and Configuring a One-Way or Two-Way LDAP Replica with Default Settings

This section discusses using default namingcontext configuration settings as a way to configure LDAP-based replication from the supplier replica to the consumer replica, with all namingcontexts included. That is, replication is configured so that the entire DIT of the supplier replica will be replicated to the consumer.

When you install and configure an LDAP replica with default settings, automated bootstrap performs the initial data synchronization from the supplier to the consumer. After the installation of a new LDAP replica is completed, LDAP based replication will be configured and the replication process will be started automatically.

Note: When you install an LDAP replica using default settings, the installer automatically migrates the consumer replica's metadata to the supplier replica. It is assumed all metadata are under the naming context cn=oraclecontext. If you need to use a different naming context, such as a realm-specific naming context, see the next section, "Installing and Configuring an LDAP-Based Replica with Customized Settings" on page 30-22.

See Also: The discussion of automatic bootstrap in Appendix H, "LDAP Replica States"

Task 1: Identify and Start the Directory Server on the Supplier Node

Identify the supplier for an LDAP-based replica. The supplier can be:

- A standalone directory
- A node of a multimaster replication group
- Another LDAP-based replica

Make sure the Oracle Internet Directory server is started on the Supplier node. To start the directory server, type the following command:

opmnctl startproc ias-component=OID
Task 2: Installing Oracle Internet Directory As An LDAP Replica

Use the instructions in "If You are Installing Oracle Internet Directory as an Advanced Replication-Based Replica or as a One-Way or Two-Way LDAP-Based Replica" on page 30-3.

- For a one-way (read-only) LDAP replica, select One-Way LDAP Replica.
- For a two-way (read-write) LDAP replica, select Two-Way LDAP Replica

When you install an LDAP replica with the default settings, Oracle Universal Installer automatically invokes bootstrap to migrate data from the supplier to the consumer.

**Note:** Bootstrapping may take a long time to complete.

Do not update the Oracle Internet Directory schema on the supplier when bootstrapping is in progress. If you do, replication bootstrap might fail. If it fails, verify that the Oracle Internet Directory schema at the consumer is synchronized with that of the supplier before you try to bootstrap again.

If you update the supplier Oracle Internet Directory during bootstrapping, the Oracle Internet Directory replication server will log a warning message. Changes you make to the supplier will be replicated to the consumer. Some of the changes, however, might be moved to the human intervention queue.

Once installation is complete, LDAP replication will be configured. The replication server on the consumer will replicate changes from the supplier.

You can check replication activity by viewing the replication log file on the new LDAP replica.

Task 3: Ensure the Directory Replication Servers are Started

After Task 2 has completed, the replication server on the new replica is automatically started. If this is a one-way replica, you do not need to do anything else. If this is a two-way replica, however, you must ensure that the replication server is started at both the sponsor and the new replica.

To start or restart the Oracle Internet Directory replication server at the sponsor replica, type:

```bash
oidctl server=oidrepld connect=connect_string_of_sponsor_replica \ 
instance=instance_number_of_consumer_replica \ 
flags="-p port_of_oid_server_running_at_sponsor_replica \ 
-h hostname_of_sponsor_replica" start
```

Installing and Configuring an LDAP-Based Replica with Customized Settings

To establish customized settings, you must first install the new node as a Master node (standalone). To do so, follow the instructions in "If You are Installing Oracle Internet Directory as a Master" on page 30-3.

After configuring LDAP base replication with remtool, you can customize the namingcontext defining what will be replicated for that LDAP-based node.

**See Also:** The discussion of naming contexts in "Determining What Is to Be Replicated in LDAP-Based Partial Replication" on page 30-33
There are two ways to install and configure an LDAP-based replica with customized setting, based on how you will migrate the data from the directory:

- Use the command-line tools. Use `ldifwrite` to backup the data from the supplier replica, then use `bulkload` to restore the data to the consumer replica
- Use automatic bootstrapping. This is a replication server feature that automatically bootstrap the data from the supplier replica to the consumer replica, based upon replication configuration.

Table 30–1 compares these two methods.

<table>
<thead>
<tr>
<th>Migration Using ldifwrite/bulkload</th>
<th>Migration Using Automatic Bootstrapping</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual procedure</td>
<td>Automatic procedure</td>
</tr>
<tr>
<td>Faster performance</td>
<td>Uses the filtering capability of partial replication</td>
</tr>
<tr>
<td>Good for a large amount of data</td>
<td>Good for a smaller number of entries</td>
</tr>
</tbody>
</table>

If automatic bootstrapping is your chosen data migration method, customize your LDAP-based replica using "Configuring an LDAP-Based Replica by Using Automatic Bootstrapping" on page 30-23.

If ldifwrite/bulkload is your chosen data migration method, configure your LDAP-based replica using "Configuring an LDAP-Based Replica by Using the ldifwrite Tool" on page 30-28.

**Configuring an LDAP-Based Replica by Using Automatic Bootstrapping**

The following eight tasks enable you to configure an LDAP-based replica by using automatic bootstrapping. They are explained in the paragraphs that follow this list.

- **Task 1: Identify and Start the Directory Server on the Supplier Node**
- **Task 2: Create the New Consumer Node by Installing Oracle Internet Directory as a Master**
- **Task 3: Back Up the Metadata from the New Consumer Node**
- **Task 4: Add an LDAP-Based Replica by Using the Replication Environment Management Tool**
- **Task 5: On the Consumer, Configure the Consumer Replica for Automatic Bootstrapping**
- **Task 6: Optional: Change Default Replication Parameters**
- **Task 7: Ensure the Directory Replication Servers are Started**
- **Task 8: If DAS or SSO Are Installed on the New Node, Restore Their Entries in the New Node's Directory**

**Task 1: Identify and Start the Directory Server on the Supplier Node** Identify the supplier for an LDAP-based replica. The supplier can be

- A standalone directory
- A node of a multimaster replication group
- Another LDAP-based replica

Make sure the Oracle Internet Directory server is started on the Supplier node. To start the directory server, type the following command:
opmnc1 startproc ias-component=OID

Task 2: Create the New Consumer Node by Installing Oracle Internet Directory as a Master To install and configure an LDAP-based replica with customized settings, you must install the new consumer node as a master. To install a new Oracle Internet Directory as a master, follow the directions in "If You are Installing Oracle Internet Directory as a Master" on page 30-3.

Task 3: Back Up the Metadata from the New Consumer Node Before configuring the new node as an LDAP-based replica with customized settings, you must first migrate its metadata to the supplier node, as follows:

- Make sure the Oracle Internet Directory server is up and running on both the supplier node and the new node created in Task 2, so that the backup process (remtool –backupmetadata) can succeed.
- From the newly created node, run the following command:

```bash
remtool –backupmetadata \
-replica "new_node_host:new_node_port/new_node_repdn_pwd" \
-master "master_host:master_port/master_repl_dn_pwd"
```

where `master_host:master_port/master_repl_dn_pwd` are the hostname, port number, and replication DN password for the desired replica’s supplier.

**Note:** If Oracle Delegated Administration Services is not configured, you might see an error message similar to this when you run `remtool` with the –backupmetadata option:

```
Failed to add "orclApplicationCommonName=ias.acme.com, cn=IAS Instances, cn=IAS, cn=Products, cn=OracleContext" as "uniqueMember" to entry "cn=Associated Mid-tiers, orclapplicationcommonname=DASApp, cn=DAS,cn=products, cn=OracleContext at replica ldap://myhost:389
```

Please ignore this error message.

**See Also:** The `remtool` command-line tool reference in Oracle Identity Management User Reference for more information about `remtool` options, including –backupmetadata.

- Apart from loading the metadata into master replica, this command creates a file named `ocbkup.new_replica_id.TO.master_replicaid.timestamp.dat` containing the metadata as back up. This file is created under the `$ORACLE_HOME/ldap/log` directory. This file contains the changes made to master replica in LDIF format, a copy of the SSO container entry [orclApplicationCommonName=ORASSO_SSOSERVER, cn=SSO, cn=Products, cn=OracleContext] and DAS URL container entry [cn=OperationURLs, cn=DAS, cn=Products, cn=OracleContext].

- If the metadata backup succeeds, it shows the following message in the terminal:

```
Backup of metadata will be stored in
$ORACLE_HOME/ldap/log/ocbkup.new_replica_id.TO.master_replicaid.timestamp.dat.
Metadata copied successfully.
```

The message will contain the path of your ORACLE_HOME.
If the metadata backup is unsuccessful, the $ORACLE_HOME/ldap/log/remtool.log file will contain error messages. If you invoked remtool from a terminal, error messages appear on that terminal.

Task 4: Add an LDAP-Based Replica by Using the Replication Environment Management Tool To add a replica, enter the following on the consumer replica:

```
remtool -paddnode [-v] [-bind supplier_host_name:port/replcation_dn_password]
```

The remtool utility will prompt for agreement type. Select One-Way or Two-Way LDAP, depending on which type of replica you are adding.

See Also: The remtool command-line tool reference in Oracle Identity Management User Reference for more information about the Replication Environment Management Tool.

Task 5: On the Consumer, Configure the Consumer Replica for Automatic Bootstrapping To use the automatic bootstrap capability, on the consumer, set the orclReplicaState attribute of the consumer replica subentry to 0 as follows:

1. Edit the sample file mod.ldif as follows:

```
Dn: orclreplicaid=unique_replicaID_of_consumer, cn=replication configuration
Changetype:modify
add:orclReplicaState
Orc1ReplicaState: 0
```

2. Use ldapmodify at the consumer to update the consumer replica's subentry orclreplicastate attribute.

```
ldapmodify -D "cn=orcladmin" -w administrator_password -h consumer_host \ -p port -f mod.ldif
```

See Also: Chapter 31, "Oracle Internet Directory Replication Monitoring and Management" for more information about the bootstrap capability of the LDAP-based replication.

Task 6: Optional: Change Default Replication Parameters You can change the default parameters for replication agreements and for the replica subentry.

See Also:
- "Replication Configuration Objects in the Directory" on page 29-8
- "Determining What Is to Be Replicated in LDAP-Based Partial Replication" on page 30-33
- "Viewing and Modifying Directory Replication Server Configuration Parameters" on page 31-1
- "Viewing and Modifying Parameters for Particular Replica Nodes" on page 31-4
- "Modifying Parameters for Replication Agreements" on page 31-6

Task 7: Ensure the Directory Replication Servers are Started The exact procedure for starting the replication servers depends on whether this is a one-way or a two-way replica.

For one-way LDAP replication, you must start the replication server at the consumer. Type:

```
oidctl server=oidrepld connect=connect_string_of_consumer_replica \ instance=instance_number_of_consumer_replica \
```
Installing and Configuring One-Way or Two-Way LDAP-Based Replication

flags= "-p port_of_oid_server_running_at_consumer \ 
-h hostname_of_sponsor_replica -m false" start

Using the \-m false option is recommended when starting the Oracle Internet Directory replication server at the consumer for one-way LDAP replication. It disables conflict resolution for better performance.

- For two-way LDAP replication, you must start the Oracle Internet Directory replication servers at both the sponsor replica and the new replica, as follows:

  1. Start or restart the replication server at the sponsor replica. Type:

     oidctl server=oidrepld connect=connect_string_of_sponsor_replica \ 
     instance=instance_number_of_sponsor_replica \ 
     flags= "-p port_of_oid_server_running_at_sponsor_replica \ 
     -h hostname_of_consumer_replica" start

  2. Start the replication server at the new replica. Type:

     oidctl server=oidrepld connect=connect_string_of_consumer_replica \ 
     instance=instance_number_of_consumer_replica \ 
     flags= "-p port_of_oid_server_running_at_new_replica \ 
     -h hostname_of_consumer_replica"

See Also: The oidctl command-line tool reference in Oracle Identity Management User Reference

When the replication server is started, it will start to bootstrap the data from the supplier to the consumer. Once the bootstrap has completed successfully, the replication server will automatically change to ONLINE mode to process changes from the supplier to the consumer.

Task 8: If DAS or SSO Are Installed on the New Node, Restore Their Entries in the New Node’s Directory

The entries for DAS and SSO must refer to the local instances of these services. However, the initial replication download from the supplier to the consumer creates these entries with values replicated from the supplier. If these services are in fact configured on the consumer node, then these values need to be replaced by the correct information appropriate to the consumer node.

- If the Delegated Administration Service (DAS) is configured on the consumer node, it must be restored using the following steps:

  1. In the ocbkup.new_replicaid.TO.master_replicaid.timestamp.dat file created by Task 3, locate and copy the DAS URL. The DN of the DAS URL container entry is "cn=OperationURLs, cn=DAS, cn=Products, cn=OracleContext". It is usually the next-to-last entry in the file.

See Also: Oracle Identity Management Guide to Delegated Administration

  2. Create an LDIF file called change_das_url.ldif with the following contents:

     dn: cn=OperationURLs,cn=DAS,cn=Products,cn=OracleContext
     changetype: modify
     replace: orcldasurlbase
     orcldasurlbase: copy_paste_the_URL_from_backup_file
3. Execute the following command to change the DAS URL:

```
ldapmodify -p consumer_port -h consumer_host -D super_user_DN \
- w super_user_password -f change_das_URL.ldif
```

Similarly, if Single Sign-on (SSO) is configured on the consumer node, it must be restored using the following steps:

1. In the `ocbkup.timestamp.dat` file created by Task 3, locate and copy the SSO container entry. Copy only the attributes shown in step 2. The DN of the SSO container entry is "orclApplicationCommonName=ORASSO_SSOSERVER, cn=SSO, cn=Products, cn=OracleContext". It is usually the last entry in the file.

See Also: Oracle Application Server Single Sign-On Administrator's Guide

2. Create an LDIF file `add_SSO_container.ldif` with the following contents:

```
dn: orclApplicationCommonName=ORASSO_SSOSERVER, 
    cn=SSO, cn=Products, cn=OracleContext
orclapplicationcommonname: ORASSO_SSOSERVER
orclappfullname: ORASSO_SSOSERVER
orclversion: 10.1.2.0.0
objectclass: orclApplicationEntity
objectclass: top
userpassword: userpassword_copied_from_backup_file
```

3. Execute the following command to add the SSO container entry:

```
ldapadd -p consumer_port -h consumer_host -D super_user_DN \
- w super_user_password -f add_SSO_container.ldif
```

4. Create an LDIF file `mod.ldif` with the following contents:

```
dn: cn=OracleUserSecurityAdmins,cn=Groups,cn=OracleContext
changetype:modify
add: uniquemember
uniquemember: orclApplicationCommonName=ORASSO_SSOSERVER, 
    cn=SSO, cn=Products, cn=OracleContext

dn: cn=verifierServices, cn=Groups,cn=OracleContext
changetype:modify
add: uniquemember
uniquemember: orclApplicationCommonName=ORASSO_SSOSERVER, 
    cn=SSO, cn=Products, cn=OracleContext
```

5. Execute the following command to apply `mod.ldif`:

```
ldapmodify -p consumer_port -h consumer_host -D super_user_DN \
- w super_user_password -f mod.ldif
```

Restart OC4J security for the change to take effect:

```
$ORACLE_HOME/opmn/bin/opmnctl stopproc process-type=OC4J_SECURITY
```
$ORACLE_HOME/opmn/bin/opmnctl startproc process-type=OC4J_SECURITY


To test Oracle Delegated Administration Services, try to log in as the admin user "orcladmin" on the Oracle Delegated Administration Services page, http(s)://new_node_hostname:new_node_http_port/oiddas/. If you cannot log in, see the troubleshooting appendix in Oracle Identity Management Guide to Delegated Administration.

To test OracleAS Single Sign-On, try to log in as the super admin user "orcladmin" on the OracleAS Single Sign-On page, http(s)://new_node_hostname:new_node_http_port/pls/orasso/. If you cannot log in, see the troubleshooting appendix in Oracle Application Server Single Sign-On Administrator’s Guide.

Configuring an LDAP-Based Replica by Using the ldifwrite Tool
This section discuss the general tasks you perform when configuring an LDAP-based replica by using the ldifwrite tool. It contains these topics:

- Task 1: Start the Directory Server on Both the Supplier and the Consumer Nodes
- Task 2: Back Up the Metadata from the New Consumer Node
- Task 3: Change the Directory Server at the Supplier to Read-Only Mode
- Task 4: Add an LDAP-Based Replica by Using the Replication Environment Management Tool
- Task 5: Back Up the Naming Contexts to Be Replicated
- Task 6: Change the Directory Server at the Supplier to Read/Write Mode
- Task 7: Load the Data on the New Consumer
- Task 8: If DAS or SSO Are Installed on the New Node, Restore Their Entries in the New Node's Directory
- Task 9: Optional: Change Default Replication Parameters
- Task 10: Ensure the Directory Replication Servers are Started

Task 1: Start the Directory Server on Both the Supplier and the Consumer Nodes

1. Identify the supplier for an LDAP-based replica. The supplier can be:
   - standalone directory
   - A node of a multi-master replication group
   - Another LDAP-based replica

   Make sure the Oracle Internet Directory server is started on the Supplier node. To start the directory server, type the following command:

   opmnctl startproc ias-component=OID

2. Identify the consumer node, which must be an new Oracle Internet Directory install as a master. To install a new Oracle Internet Directory as a Master, follow the directions in "If You are Installing Oracle Internet Directory as a Master" on page 30-3. Make sure the Oracle Internet Directory server is started on the new consumer node. To start the directory server, type the following command:

   opmnctl startproc ias-component=OID
Task 2: Back Up the Metadata from the New Consumer Node Before configuring the consumer as an LDAP-based replica with customized settings, you must first migrate its metadata to the supplier node, as follows:

- Make sure the Oracle Internet Directory server is up and running on both the supplier node and the new node created in Task 2, so that the backup process (remtool -backupmetadata) can succeed.

- From the consumer node, run the following command:
  
  ```
  remtool -backupmetadata \
  -replica "consumer_host:consumer_port/consumer_repl_dn_pwd" \
  -master "supplier_host:supplier_port/supplier_repl_dn_pwd"
  ```

- Apart from loading the metadata into master replica, this tool creates a file named `ocbkup.consumer_replica_id.TO.supplier_replica_id.timestamp.dat` containing the metadata as backup. This file is created in the `$ORACLE_HOME/ldap/log` directory. This file contains the changes made to the master replica in LDIF format, a copy of SSO container entry `[orclApplicationCommonName=ORASSO_SSOSERVER, cn=SSO, cn=Products, cn=OracleContext]` and DAS URL container entry `[cn=OperationURLs, cn=DAS, cn=Products, cn=OracleContext]`.

- If the metadata backup succeeded, `remtool` displays the following message in the terminal:
  
  ```
  Backup of metadata will be stored in
  $ORACLE_HOME/ldap/log/ocbkup.consumer_replica_id.TO.supplier_replica_id.timestamp.dat.
  Metadata copied successfully.
  ```

  The message will contain the actual path of your ORACLE_HOME and filename.

  If the metadata backup is unsuccessful, the `$ORACLE_HOME/ldap/log/remtool.log` file will contain error messages. If you invoked `remtool` from a terminal, error messages appear on that terminal.

Task 3: Change the Directory Server at the Supplier to Read-Only Mode To ensure data consistency, change the directory server on the supplier node to read-only. To do this:

1. Create an LDIF file containing the following:
   
   ```
   Dn:
   ChangeType: modify
   Replace: orclservermode
   Tclservermode: r
   ```

2. On the supplier, run the following command:
   
   ```
   ldapmodify -D "cn=orcladmin" -w administrator_password \
   -h host_name_of_supplier_node -p port -f name_of_LDIF_file.ldif
   ```

Task 4: Add an LDAP-Based Replica by Using the Replication Environment Management Tool To add a replica, enter the following on the consumer replica:

```
remtool -paddnode [-v] [-bind supplier_host_name:port/replication_dn_password]
```

The remtool utility will prompt for agreement type. Select One-Way or Two-Way LDAP, depending on which type of replica you are adding.
Task 5: Back Up the Naming Contexts to Be Replicated  If there is a large number of entries in the naming contexts that you want to replicate to the LDAP-based replica, then Oracle Corporation recommends that you back up these naming contexts at the supplier node and then load them to the LDAP-based replica.

To back up the naming contexts:

1. Identify the replication agreement DN created in "Task 4: Add an LDAP-Based Replica by Using the Replication Environment Management Tool" on page 30-29.
   
   
   ```bash
   ldapsearch -h supplier_host -p port \\
   -b "orclreplicaid=supplier_replicaID, cn=replication configuration" \\
   -s sub "(orclreplicadn= orclreplicaid=consumer_replica_ID, \\
   cn=replication configuration)" dn
   ```

2. On the supplier, use the following command to get the data from the supplier. Data loaded into the file will be based on the agreement configured:

   ```bash
   ldifwrite connect="connect_string_of_sponsor_node" \\
   basedn="replication_agreement_dn_retrieved_in_step_1" \\
   file="name_of_output_LDIF_file"
   ```

See Also:

"Determining What Is to Be Replicated in LDAP-Based Partial Replication" on page 30-33

The ldifwrite command-line tool reference in Oracle Identity Management User Reference for more instructions on using ldifwrite to back up part of the naming context

Task 6: Change the Directory Server at the Supplier to Read/Write Mode  If you performed "Task 3: Change the Directory Server at the Supplier to Read-Only Mode" on page 30-29, then change the directory server on the supplier back to read/write mode.

To do this:

1. Create an LDIF file containing the following:

   ```
   Dn:
   Changetype: modify
   Replace: orclservermode
   Orclservermode: rw
   ```

2. Run the following command:

   ```bash
   ldapmodify -D "cn=orcladmin" -w administrator_password \\
   -h host_name_of_supplier_node -p port -f name_of_LDIF_file
   ```

Task 7: Load the Data on the New Consumer  To do this:

1. If there are multiple files, then combine them into one file—for example, backup_data.ldif.

2. If naming contexts exist on the LDAP-based consumer replica, then remove them by using bulkdelete. Enter the following:

   ```bash
   bulkdelete connect="connect_string_of_replica" basedn="naming_context"
   ```
Perform this step for each naming context that was backed up in "Task 5: Back Up the Naming Contexts to Be Replicated" on page 30-30.

On the consumer, load the data to the replica by using bulkload in the append mode. Enter the following:

```
bulkload connect="connect_string_of_replica" append="TRUE" check="TRUE" \
generate="TRUE" restore="TRUE" file="backup_data.ldif"
bulkload connect="connect_string_of_replica" load="TRUE"
```

**Note:** If you load data from an earlier version of Oracle Internet Directory, such as 10g Release 2 (10.1.2.0.2) onto a node running 10g (10.1.4.0.1), you must update the password policy entries as described in "Password Policy and Fan-out Replication" on page 30-32.

---

**Task 8: If DAS or SSO Are Installed on the New Node, Restore Their Entries in the New Node's Directory** Follow the procedure described in "Task 8: If DAS or SSO Are Installed on the New Node, Restore Their Entries in the New Node's Directory" on page 30-26.

**Task 9: Optional: Change Default Replication Parameters** You can change the default parameters for replication agreements, for the replica subentry, and for the replication naming context configuration objects.

---

**Task 10: Ensure the Directory Replication Servers are Started** The exact procedure for starting the replication servers depends on whether this is a one-way or a two-way replica.

- For one-way LDAP replication, you must start the replication server at the consumer. Type:

  ```
  oidctl server=oidrepld connect=connect_string_of_consumer_replica \ 
  instance=instance_number_of_consumer_replica \ 
  flags= "-p port_of_oid_server_running_at_consumer_replica \ 
      -h hostname_of_consumer_replica -m false" start
  ```
Using the \texttt{-m false} option is recommended when starting the Oracle Internet Directory replication server at the consumer for one-way LDAP replication. It disables conflict resolution for better performance.

- For two-way LDAP replication, you must start the Oracle Internet Directory replication servers at both the sponsor replica and the new replica, as follows:

1. Start or restart the replication server at the sponsor replica. Type:

   \begin{verbatim}
   oidctl server=oidrepld connect=connect_string_of_sponsor_replica \
   instance=instance_number_of_sponsor_replica \ 
   flags="-p port_of_oid_server_running_at_sponsor_replica \
   -h hostname_of_sponsor_replica" start
   \end{verbatim}

2. Start the replication server at the new replica. Type:

   \begin{verbatim}
   oidctl server=oidrepld connect=connect_string_of_consumer_replica \
   instance=instance_number_of_consumer_replica \ 
   flags="-p port_of_oid_server_running_at_new_replica \
   -h hostname_of_consumer_replica"
   \end{verbatim}

\textbf{See Also:} The \texttt{oidctl} command-line tool reference in \textit{Oracle Identity Management User Reference}.

### Password Policy and Fan-out Replication

In 10g (10.1.4.0.1), Oracle Internet Directory supports more fine-grained password policies than in previous releases. This can cause problems when a master node is an earlier version, such as 10g Release 2 (10.1.2.0.2), and the new fan-out node is 10g (10.1.4.0.1). The password policy entries from the master node might not behave as expected when bootstrapped or migrated to the fan-out node. To correct this problem, as part of the fan-out node setup, after the data from the master is bootstrapped or migrated, you must update the password policy entries by invoking the Java class \texttt{oracle.ldap.oidinstall.backend.OIDUpgradePasswordPolicies}. The command-line syntax is as follows:

\begin{verbatim}
\end{verbatim}

Table 30–2 describes the command-line parameters.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>host</td>
<td>The host on which the 10g (10.1.4.0.1) directory server is running</td>
</tr>
<tr>
<td>port</td>
<td>The port on which the 10g (10.1.4.0.1) directory server is listening. If the protocol is \texttt{ssl}, \texttt{port} must be an SSL port.</td>
</tr>
<tr>
<td>bindDN</td>
<td>The privileged administrative user, usually \texttt{cn=orcladmin}</td>
</tr>
<tr>
<td>bindPwd</td>
<td>The user password associated with the bindDN</td>
</tr>
<tr>
<td>oracle_home</td>
<td>The Oracle home for this instance of Oracle Internet Directory</td>
</tr>
<tr>
<td>protocol</td>
<td>An optional parameter. It should be \texttt{ssl} in an SSL environment</td>
</tr>
</tbody>
</table>

All actions performed by the tool are logged to \texttt{ppUpgrade.log} in the \texttt{oracle_home/ldap/log} directory.
Deleting an LDAP-Based Replica

This section explains how to delete an LDAP-based replica. It contains these topics:

- Task 1: Stop the Directory Replication Server on the Node to be Deleted
- Task 2: Delete the Replica from the Replication Group
- Task 3: Stop the Directory Server on the Node to be Deleted

**Note:** Before running the tool, ensure that the appropriate environment variable is set correctly:

- On Linux or Solaris, the `LD_LIBRARY_PATH` variable must include `oracle_home/lib` and `oracle_home/network/lib`.
- On 64-bit Solaris, the `LD_LIBRARY_PATH` variable must include `oracle_home/lib32` and `oracle_home/network/lib32`.
- On Windows, the `PATH` variable must include `oracle_home/bin` and `oracle_home/network/bin`.

**See Also:** Chapter 19, "Password Policies in Oracle Internet Directory"

**Deleting an LDAP-Based Replica**

This section explains how to delete an LDAP-based replica. It contains these topics:

- Task 1: Stop the Directory Replication Server on the Node to be Deleted
- Task 2: Delete the Replica from the Replication Group
- Task 3: Stop the Directory Server on the Node to be Deleted

**Note:** You cannot delete a replica if it is a supplier for another replica. To delete such a replica, you must first delete all its consumers from the replication group.

**Task 1: Stop the Directory Replication Server on the Node to be Deleted**

Stop the Oracle directory replication server, following the procedure described in the `oidctl` command-line tool reference in Oracle Identity Management User Reference.

**Task 2: Delete the Replica from the Replication Group**

Do this by using the Replication Environment Management Tool. Enter:

```
remtool -pdelnode [-v] [-bind hostname:port_number/replication_dn_password]
```

**See Also:** The `remtool` command-line tool reference in Oracle Identity Management User Reference

**Task 3: Stop the Directory Server on the Node to be Deleted**

**See Also:** The `oidctl` command-line tool reference in Oracle Identity Management User Reference

**Determining What Is to Be Replicated in LDAP-Based Partial Replication**

In LDAP-based partial replication, you can determine what is or is not replicated by defining replica naming context objects. The parameters for these objects are stored in entries that have this DN:

```
cn=namingcontext_ID,cn=replication namecontext,
orclAgreementID=numeric_identifier_of_replication_agreement,
orclReplicaId=unique_identifier_of_replica, cn=replication configuration
```
Viewing and Modifying Replica Naming Context Objects by Using Oracle Directory Manager

To view and modify parameters for replica naming context objects:

1. In the navigator pane, expand in succession Oracle Internet Directory Servers, directory server instance, Replication Management, Replica Node: replica identifier, Replica Agreement: replication agreement identifier.

2. Select the replica naming context you want to modify. The Replica Naming Context tab page appears in the right pane. The fields in this tab page are described in Table A–24, "Fields in the Replica Agreement: Replica Naming Context Tab Page".

3. After you have entered the appropriate information, choose OK.

Adding Replica Naming Context Objects by Using Oracle Directory Manager

1. In the navigator pane, expand in succession Oracle Internet Directory Servers, directory server instance, Replication Management, Replica Node: replica identifier, Replica Agreement: replication agreement identifier.

2. Select Naming Context:naming context identifier.

3. From the toolbar, choose Create. The New Replica Agreement Naming Context dialog box appears.

4. In the fields in the New Replica Agreement Naming Context dialog box, enter the appropriate information. The fields in this dialog box are described in Table A–24, "Fields in the Replica Agreement: Replica Naming Context Tab Page".

5. Choose OK.

Deleting Replica Naming Context Objects by Using Oracle Directory Manager

1. In the navigator pane, expand in succession Oracle Internet Directory Servers, directory server instance, Replication Management, Replica Node: replica identifier, Replica Agreement: replication agreement identifier.

2. Using your mouse, right-click Naming Context:naming context identifier.

3. Select Delete.

Modifying Replica Naming Context Object Parameters by Using ldapmodify

Replica naming context object parameters are listed and described under Replication Schema Elements in Oracle Identity Management User Reference.

**Note:** The replication server reads naming context objects from the supplier replica.

Example 30–1 Adding a Naming Context Object for an LDAP-Based Replica

This example creates a naming context object that does the following:

---

**Note:** Because the directory replication server reads replica naming context objects from the agreement located at the supplier, you must apply all modifications against naming context objects at the supplier and, optionally, at the consumer.
Installing and Configuring One-Way or Two-Way LDAP-Based Replication

- Replicates the naming context `ou=Americas, cn=mycompany`
- Excludes from replication the naming context `cn=customer profile, ou=Americas, cn=mycompany`
- Excludes from replication the attribute `userpassword`

The steps are:

1. Edit the example file `mod.ldif` as follows:

   ```
   dn: cn=naming_context_identifier, cn=replication namecontext, 
   orclagreementid=replication_agreement_identifier, 
   orclreplicaid=supplier_replica_identifier, cn=replication configuration 
   orclincludednamingcontexts: ou=Americas, cn=mycompany 
   orclexcludednamingcontexts: cn=customer profile, ou=Americas, cn=mycompany 
   orclexcludedattributes: userpassword 
   objectclass: top 
   objectclass: orclreplnamectxconfig 
   ```

2. Use `ldapadd` to add the partial replication naming context object to the supplier.

   ```
   ldapadd -D "cn=orcladmin" -w administrator_password -h supplier_host 
   -p port_number -f mod.ldif 
   ```

**Example 30–2  Deleting a Naming Context Object**

To delete the naming context object created in Example 30–1, type:

```
ldapdelete -D "cn=orcladmin" -w administrator_password -h supplier_host 
-h supplier_host_port_number 
*"cn=naming_context_identifier, cn=replication namecontext, 
orclagreementid=replication_agreement_identifier, 
orclreplicaid=supplier_replica_identifier, 
 cn=replication configuration"*
```

**Example 30–3  Modifying the orclIncludedNamingContexts Attribute for a Replica Naming Context Object**

The directory replication server uses the `orclIncludedNamingContexts` attribute value of the replica naming context object to specify the top-level subtree included in partial replication.

In this example, the included naming context is set to `c=us`, which means that `c=us` is to be included in partial replication.

1. Edit the example file `mod.ldif` as follows:

   ```
   DN: cn=naming_context_identifier, cn=replication namecontext, 
   orclagreementid=replication_agreement_identifier, 
   orclreplicaid=supplier_replica_identifier, cn=replication configuration 
   Changetype:modify 
   Replace: orclIncludedNamingcontexts 
   orclIncludedNamingcontexts: c=us 
   ```

2. Use `ldapmodify` to update the replication agreement `orclupdateschedule` attribute.

   ```
   ldapmodify -D "cn=orcladmin" -w administrator_password -h supplier_host 
   -p port -f mod.ldif 
   ```

3. Restart the directory replication server.
Example 30–4  Modifying the orclExcludedNamingContexts Attribute for a Replica Naming Context Object

The directory replication server uses the orclExcludedNamingcontext attribute value of the replica naming context object to specify the top-level subtrees excluded from partial replication.

In this example, the excluded naming contexts are set to ou=Europe,c=us and ou=Americas,c=us, which means that these two naming contexts are to be excluded from partial replication.

1. Edit the example file mod.ldif as follows:

   DN:cn=naming_context_identifier,
   cn=replication namecontext,
   orclagreementid=replication_agreement_identifier,
   orclreplicaid=supplier_replica_identifier,cn=replication configuration
   Changetype:modify
   Replace: orclExcludedNamingcontexts
   orclExcludedNamingcontexts: ou=Europe, c=us
   orclExcludedNamingcontexts: ou=Americas, c=us

2. Use ldapmodify to update the replication agreement orclupdateschedule attribute.

   ldapmodify -D "cn=orcladmin" -w administrator_password \
   \-h supplier_host -p port -f mod.ldif

3. Restart the directory replication server.

   Note: A subtree specified in the orclExcludedNamingcontexts attribute must also be a subtree of the specified includedNamingcontext of the same replica naming context object.

Example 30–5  Modifying the orclExcludedAttributes Attribute for a Replica Naming Context Object

You can specify that certain changes made to the included naming context be excluded, at attribute level, from partial replication. To determine which attributes are to be excluded, the directory replication server uses the value of the orclExcludedAttributes attribute of the replica naming context object.

In this example, the telephonenumber and title attributes of the naming context specified in the orclincludedNamingcontexts attribute are excluded from replication.

1. Edit the example file mod.ldif as follows:

   DN:cn=naming_context_identifier,
   cn=replication namecontext,
   orclagreementid=replication_agreement_identifier,
   orclreplicaid=supplier_replica_identifier,cn=replication configuration
   Changetype:modify
   Replace: orclExcludedAttributes
   orclExcludedAttributes: telephonenumber
   orclExcludedAttributes: title

2. Use ldapmodify to update the replication agreement orclupdateschedule attribute.

   ldapmodify -D "cn=orcladmin" -w administrator_password -h my_host \
   -p port -f mod.ldif

3. Restart the directory replication server.
Resolving Conflicts Manually in a Replication Group

This section contains these topics:

- Monitoring Replication Change Conflicts
- Examples of Conflict Resolution Messages
- About the Human Intervention Queue Manipulation Tool
- About the Oracle Internet Directory Comparison and Reconciliation Tool

Monitoring Replication Change Conflicts

If a conflict has been written into the log, then it means that the system is not able to resolve it by following its resolution procedure. To avoid further replication change conflicts arising from earlier unapplied changes, it is important to monitor the logs regularly.

To monitor replication change conflicts, examine the contents of the replication log. You can distinguish between messages by their respective timestamps.

Examples of Conflict Resolution Messages

Conflict resolution messages, examples of which are shown in this section, are logged in the file `oidrepld00.log`. The path for this file is `$ORACLE_HOME/ldap/log`. The result of each attempt to resolve the replication conflict is displayed at the end of each conflict resolution message.

Example 30–6 An Attempt to Modify a Non-Existent Entry

```
2000/08/03::10:59:05:  ************ Conflict Resolution Message ************
2000/08/03::10:59:05:  Conflict reason: Attempted to modify a non-existent entry.
2000/08/03::10:59:05:  Change number:1306.
2000/08/03::10:59:05:  Change type:Modify.
2000/08/03::10:59:05:  Target DN:cn=ccc,ou=Recruiting,ou=HR,ou=Americas,o=IMC,c=US.
2000/08/03::10:59:05:  Result: Change moved to low priority queue after failing on 10th retry.
```

Example 30–7 An Attempt to Add an Existing Entry

```
2000/08/03::10:59:05:  ************ Conflict Resolution Message ************
2000/08/03::10:59:05:  Conflict reason: Attempted to add an existing entry.
2000/08/03::10:59:05:  Change number:1209.
2000/08/03::10:59:05:  Change type:Add.
2000/08/03::10:59:05:  Target DN:cn=Lou Smith, ou=Recruiting, ou=HR, ou=Americas, o=IMC, c=US.
2000/08/03::10:59:05:  Result: Deleted duplicated target entry which was created later than the change entry. Apply the change entry again.
```

Example 30–8 An Attempt to Delete a Non-Existent Entry

```
2000/08/03::10:59:06:  ************ Conflict Resolution Message ************
2000/08/03::10:59:06:  Conflict reason: Attempted to delete a non-existent entry.
2000/08/03::10:59:06:  Change number:1365.
2000/08/03::10:59:06:  Change type:Delete.
2000/08/03::10:59:06:  Target DN:cn=Lou
About the Human Intervention Queue Manipulation Tool

The Human Intervention Queue Manipulation Tool enables you to move changes from the human intervention queue to either the retry queue or the purge queue. Moving the change to the purge queue means that there are no further attempts to re-apply the changelog entry. To address changes in the human intervention queue, follow these general steps:

1. Shut down the directory replication server.
2. Analyze the replication log.
3. Use the Human Intervention Queue Manipulation Tool to move the changes to either the retry queue or the purge queue as described in the following sections.

---

**Note:** The Oracle Internet Directory server parameter `orclSizeLimit`, which is 1000 by default, limits the number of entries that the Human Intervention Queue Manipulation Tool can process. If you have more than 1000 entries in the human intervention queue, you must increase `orclSizeLimit`, or some entries will never be processed. Setting the parameter `orclSizeLimit` very high will impact server performance, because `orclSizeLimit` also controls the maximum number of entries to be returned by a search.

---

**See Also:** The `hiqretry.sh` command-line tool reference in *Oracle Identity Management User Reference* for instructions on how to use the Human Intervention Queue Manipulation Tool.

About the Oracle Internet Directory Comparison and Reconciliation Tool

When the directory replication server encounters inconsistent data, you can use the Oracle Internet Directory Comparison and Reconciliation Tool to synchronize the entries on the consumer with those on the supplier. When you do this, perform the following general steps:

1. Set the supplier and the consumer to read-only mode.
2. Ensure that the supplier and the consumer are in a tranquil state—that is, that neither is supplying or applying changes. If they are not in a tranquil state, then wait until they have finished updating.
3. Identify the inconsistent entries or subtree on the consumer.
4. Use the Oracle Internet Directory Comparison and Reconciliation Tool to fix the inconsistent entries or subtree on the consumer.
5. Set the participating supplier and consumer back to read/write mode.
Example: Installing and Configuring a Multimaster Replication Group with Fan-Out

To help you install and configure a multimaster replication group with fan-out, this section offers an example with four systems as described in Table 30–3.

Table 30–3 Nodes in Example of Partial Replication Deployment

<table>
<thead>
<tr>
<th>Node</th>
<th>Host Name</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node1</td>
<td>mycompany1.com</td>
<td>3000</td>
</tr>
<tr>
<td>Node2</td>
<td>mycompany2.com</td>
<td>4000</td>
</tr>
<tr>
<td>Node3</td>
<td>mycompany3.com</td>
<td>5000</td>
</tr>
<tr>
<td>Node4</td>
<td>mycompany4.com</td>
<td>6000</td>
</tr>
<tr>
<td>Node5</td>
<td>mycompany5.com</td>
<td>7000</td>
</tr>
</tbody>
</table>

In this example, the user has set up the following requirements:

- **Requirement 1**: Node1 and Node2 must be synchronized so that changes made on either node are replicated to the other— but the naming context cn=private users, cn=mycompany is to be excluded from this replication.

- **Requirement 2**: The naming context ou=Americas, cn=mycompany on node3 is to be partially synchronized from Node2 so that only changes made under ou=Americas, cn=mycompany on Node2 are replicated to Node3. The following are to be excluded from this replication:
  - Changes made under cn=customer profile, ou=Americas, cn=mycompany
  - Changes in the attribute userpassword.

- **Requirement 3**: Node4 is to be configured as a full replica of node 2, that is, changes to all naming contexts in Node2 will be replicated (one-way) to Node4.

- **Requirement 4**: Node5 is to be configured as a two-way (updatable) full replica of Node1.

See Also:

- "Task 3: Switch the Sponsor Node to Read-Only Mode" for instructions on setting a node to read-only mode
- The oidcmprec command-line tool reference in Oracle Identity Management User Reference for syntax and an explanation of how Oracle Internet Directory Comparison and Reconciliation Tool works.
To meet the first requirement in this example, we set up a multimaster replication group for Node1 and Node2. To meet the second, we set up a partial replica for Node2 and Node3, and for the third, full LDAP replication from Node2 to Node4.

This section contains these topics:

- Task 1: Set up the Multimaster Replication Group for Node1 and Node2
- Task 2: Configure the Replication Agreement
- Task 3: Start the Replication Servers on Node1 and Node2
- Task 4: Test the Directory Replication Between Node1 and Node2
- Task 5: Install and Configure Node3 as a Partial Replica of Node2
- Task 6: Customize the Partial Replication Agreement
- Task 7: Start the Replication Servers on All Nodes in the DRG
- Task 8: Install and Configure Node4 as a Full Replica of Node2
- Task 9: Test the Replication from Node2 to Node4
- Task 10: Install and Configure Node5 as a Two-Way Replica of Node1
- Task 11: Test the Two-Way Replication Between Node1 and Node5

**Task 1: Set up the Multimaster Replication Group for Node1 and Node2**

To set up the multimaster replication group for Node1 and Node2, follow Tasks 1 through 5 in "Installing and Configuring a Multimaster Replication Group" on page 30-6.

**Task 2: Configure the Replication Agreement**

In the replication agreement between Node1 and Node2, specify the value for the orclExcludedNamingcontexts attribute as cn=private users,cn=mycompany. To do this:

1. Edit the example file mod.ldif as follows:

   ```
   dn: orclAgreementID=000001,cn=replication configuration
   Changetype:modify
   Replace: orclExcludedNamingcontexts
   orclExcludedNamingcontexts: cn=private users,cn=mycompany
   ```
2. Use ldapmodify to update the replication agreement orclExcludedNamingcontexts attribute at both Node1 and Node2. To do this, enter:

ldapmodify -D "cn=orcladmin" -w administrator_password -h mycompany1.com -p 3000 -f mod.ldif
ldapmodify -D "cn=orcladmin" -w administrator_password -h mycompany2.com -p 4000 -f mod.ldif

Task 3: Start the Replication Servers on Node1 and Node2
To do this, follow the instructions in "Task 6: Start the Replication Servers on All Nodes in the DRG" on page 30-13.

Task 4: Test the Directory Replication Between Node1 and Node2.
To do this, follow the instructions in "Task 7: Test Directory Replication" on page 30-14.

Task 5: Install and Configure Node3 as a Partial Replica of Node2
If you want to use the bootstrap capability of partial replication, then follow Tasks 1 through 5 in "Configuring an LDAP-Based Replica by Using Automatic Bootstrapping" on page 30-23.

If you want to configure the replica by using the ldifwrite tool, then follow Tasks 1 through 9 in "Configuring an LDAP-Based Replica by Using the ldifwrite Tool" on page 30-28.

Identify Node2 as the supplier and Node3 as the consumer.

Task 6: Customize the Partial Replication Agreement
To do this:

- To achieve Requirement 2 in this example, the default replication between Node2 and Node3 must be configured first:

  In partial replication, the cn=oraclecontext naming context is replicated by default. You can choose not to replicate it by deleting it at both the supplier (Node2, mycompany2.com) and the consumer (Node3, mycompany3.com).

  ldapdelete -D "cn=orcladmin" -w administrator_password -h mycompany2.com -p 4000 "cn=includednamingcontext000001, cn=replication namecontext,orclagreementid=000002, orclreplicaid=node2_replica_id, cn=repl..."

  ldapdelete -D "cn=orcladmin" -w administrator_password -h mycompany3.com -p 5000 "cn=includednamingcontext000001, cn=replication namecontext,orclagreementid=000002, orclreplicaid=node2_replica_id, cn=repl..."

- To replicate the naming context ou=Americas, cn=mycompany, and to exclude from replication the naming context cn=customer profile, ou=Americas, cn=mycompany and the attribute userpassword, create a naming context object as follows:

  a. Edit the example file mod.ldif as follows:

   
   dn: cn=includednamingcontext000002,cn=replication namecontext,orclagreementid=000002,orclreplicaid=node2_replica_id, cn=repl...
Example: Installing and Configuring a Multimaster Replication Group with Fan-Out

```oracli
orclincludednamingcontexts: ou=Americas,cn=mycompany
orclexcludednamingcontexts: cn=customer profile, ou=Americas, cn=mycompany
orclexcludedattributes: userpassword
objectclass: top
objectclass: orclreplnamectxconfig
```

b. Use `ldapadd` to add the partial replication naming context object at both Node2 and Node3.

```bash
ldapadd -D "cn=orcladmin" -w administrator_password -h mycompany2.com -p 4000 -f mod.ldif
ldapadd -D "cn=orcladmin" -w administrator_password -h mycompany3.com -p 5000 -f mod.ldif
```

- If you decide to use the automatic bootstrap capability of partial replication, then edit the example file `mod.ldif` and use `ldapmodify` to modify the partial replica `orclreplicastate` attribute at both Node2 and Node3, as described in "Task 5: On the Consumer, Configure the Consumer Replica for Automatic Bootstrapping" on page 30-25.

**Task 7: Start the Replication Servers on All Nodes in the DRG**

To do this, follow the instructions in "Task 10: Ensure the Directory Replication Servers are Started" on page 30-31.

**Task 8: Install and Configure Node4 as a Full Replica of Node2**

Since full replica replication is the default configuration when the new node is installed as an LDAP replica, use the instructions in "Installing and Configuring a One-Way or Two-Way LDAP Replica with Default Settings" on page 30-21. When the installer prompts for the supplier information, provide the supplier hostname, `mycompany2.com`, the supplier port, 4000, and the super user password.

**Note:** While installing node 4 as described in "If You are Installing Oracle Internet Directory as an Advanced Replication-Based Replica or as a One-Way or Two-Way LDAP-Based Replica" on page 30-3, you will be asked to provide the supplier's hostname, port, and password of the super user cn=orcladmin.

**Task 9: Test the Replication from Node2 to Node4**

Test this replication using the instructions in the section entitled "Task 7: Test Directory Replication" on page 30-14.

**Task 10: Install and Configure Node5 as a Two-Way Replica of Node1**

Because full replica replication is the default configuration when the new node is installed as an LDAP replica, follow the instructions in "Installing and Configuring a One-Way or Two-Way LDAP Replica with Default Settings" on page 30-21. When the installer prompts for the supplier information, provide the supplier hostname, `mycompany1.com`, the supplier port, 3000, and the super user password.

**Task 11: Test the Two-Way Replication Between Node1 and Node5**

Create an entry at Node1 using either Oracle Directory Manager or the `ldapadd` command-line tool. Wait for the entry to be replicated at Node5. After the entry is replicated to Node5, apply a change to that entry at Node5 using either the `ldapmodify` command-line tool or Oracle Directory Manager. The change will be replicated to Node1.
Configuring Replication Failover

This section contains the following topics:

- Limitations and Warnings for Replication Failover
- Determining Which Type of Replication Failover to Use
- Performing a Stateless Replication Failover
- Performing a Time-Based Replication Failover

Limitations and Warnings for Replication Failover

This section describes limitations and warnings related to the use of replication failover.

- As of Oracle Internet Directory 10g (10.1.4.0.1), replication failover requires administrator intervention.
- Following failover, you must compare and reconcile the consumer with the new supplier.
- The new agreement must be of the same type and direction as the old agreement.
- Only two topology types are supported. See "Replication Failover" on page 29-27.
- When a supplier fails, its directly connected replica can only fail over to another directly connected replica of the failed supplier.
- The replication filtering policy for the agreement between the new supplier and old supplier must match that between the old supplier and consumer.
- In most cases, you should fail over the replica in a way that preserves the original replica type. In the case shown in Figure 30–2, node 2 is the old supplier for both node 1 and 3, and node 1 is read-only. When node 2 fails, you could, in theory, set up either node 1 or 3 as the new supplier node. Best practice, however, is to fail over node 1 so that node 3 is the supplier. This preserves node 1’s original, read-only replica type.

Figure 30–2  Failover Preserving Replica Type

If the new agreement is a two-way agreement, after you compare and reconcile the consumer with its new supplier, you must also compare and reconcile all other replicas that are connected to the new supplier with the new supplier. For example, in Figure 30–3, Node 2 has a two-way agreement with Node 3. Node 3 is connected to another replica, Node 4. When Node 2 fails, you set up a two-way agreement between node 3 and node 1. After comparing and reconciling node 3 with node 1, you must also compare and reconcile Node 4 with node 3 to ensure that the replicas are synchronized.
Determining Which Type of Replication Failover to Use

There are two types of replication failover. They are:

- Stateless
- Time-based

Use stateless failover when you are unable to plan for the failover in advance. Stateless replication failover makes no assumptions about the state of the replicas. You can fail over to a new supplier at any time. Stateless failover requires more work after failover to synchronize the nodes.

Use time-based failover for planned failover. Time-based failover results in less work after failover. However, it requires some setup ahead of time to ensure that the following assumptions are true at the time of failover:

- The nodes are mostly synchronized
- The new supplier has preserved its change logs so that complete synchronization can be achieved quickly.

Performing a Stateless Replication Failover

This section explains how to perform a stateless replication failover. It consists of the following tasks:

- Task 1: Stop all Directory Replication Server on related Nodes
- Task 2: Break Old Replication Agreement and Setup New Agreement
- Task 3: Save Last Change Number
- Task 4: Compare and Reconcile New Supplier and Consumer
- Task 5: Update Last Applied Change Number of New Agreement
- Task 6: Clean Up Old Agreement on Old Supplier
- Task 7: Start all Directory Replication Server on related Nodes

Task 1: Stop all Directory Replication Server on related Nodes

Stop the Oracle directory replication servers on the new supplier, old supplier and consumer, by typing:

```
oidctl connect=db_connect_string server=oidrepld instance=instance_number \
    host=hostname stop
```
Configuring Replication Failover

Task 2: Break Old Replication Agreement and Setup New Agreement

Break the old replication agreement between the old supplier and consumer and set up a new agreement between the new supplier and consumer. Do this by using the Replication Environment Management Tool. Type:

```
remtool -pchgmaster [-v] [-bind consumer_host::port_number/replication_dn_password]
```

See Also: The remtool command-line tool reference in Oracle Identity Management User Reference.

Task 3: Save Last Change Number

Obtain the last change number from the new supplier.

For a one-way agreement, use the following command:

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

For a two-way agreement, use the following command:

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

Save this number!

Task 4: Compare and Reconcile New Supplier and Consumer

Use the Oracle Internet Directory Comparison and Reconciliation Tool to compare and reconcile the new supplier and consumer. For a one-way agreement, type:

```
oidcmprec operation=reconcile ""
   source=new_supplier_host:port/new_supplier_replication_dn_passwd
   destination=consumer_host:port/consumer_replication_dn_passwd
   base="" scope=sub
```

For a two-way agreement, type:

```
oidcmprec operation=merge ""
   source=new_supplier_host:port/new_supplier_replication_dn_passwd
   destination=consumer_host:port/consumer_replication_dn_passwd
   base="" scope=sub
```

This example assumes that the entire directory is replicated and, therefore, that base is set to "". If you are using partial replication, use the base and dns2exclude arguments to the oidcmprec tool to include the desired DIT.

See Also: The oidcmprec command-line tool reference in Oracle Identity Management User Reference.
Task 5: Update Last Applied Change Number of New Agreement

Modify the new agreement with the retrieved last applied number at the new supplier. To do this:

1. Create an LDIF file with the last change number you retrieved in "Task 3: Save Last Change Number".

   For a one-way agreement, it should look similar to this:

   ```
   dn: agreement_dn
   changetype: modify
   replace: orclLastAppliedChangeNumber;apply$new_supplier_host$consumer_host
   orclLastAppliedChangeNumber;apply$new_supplier_host$consumer_host:
       last_change_number_retrieved.
   -replace: orclLastAppliedChangeNumber;transport$new_supplier_host$consumer_host
   orclLastAppliedChangeNumber;transport$new_supplier_host$consumer_host:
       last_change_number_retrieved_from_new_supplier
   ```

   For a two-way agreement, it should look similar to this:

   ```
   dn: agreement_dn
   changetype: modify
   replace: orclLastAppliedChangeNumber;apply$new_supplier_host$consumer_host
   orclLastAppliedChangeNumber;apply$new_supplier_host$consumer_host:
       last_change_number_retrieved_from_new_supplier
   -replace: orclLastAppliedChangeNumber;transport$new_supplier$consumer
   orclLastAppliedChangeNumber;transport$new_supplier$consumer:
       last_change_number_retrieved_from_new_supplier
   -replace: orclLastAppliedChangeNumber;apply$consumer_host$new_supplier_host
   orclLastAppliedChangeNumber;apply$consumer_host$new_supplier_host:
       last_change_number_retrieved_from_consumer
   -replace: orclLastAppliedChangeNumber;transport$consumer_host$new_supplier_host
   orclLastAppliedChangeNumber;transport$consumer_host$new_supplier_host:
       last_change_number_retrieved_from_consumer
   ```

2. Modify the agreement by using `ldapmodify`, as follows:

   ```
   ldapmodify -D "cn=orcladmin" -w password -h host_name -p port_number \
   -f LDIF_file
   ```

Task 6: Clean Up Old Agreement on Old Supplier

If the old supplier was down when you performed "Task 2: Break Old Replication Agreement and Setup New Agreement", the old agreement on the old supplier was not cleaned up. Clean it up now by using the Replication Environment Management Tool. Type:

```remtool -p cleanup -agrmt [-v] [-bind consumer_host::port_number/replication_dn_password]```

See Also: The `remtool` command-line tool reference in *Oracle Identity Management User Reference*
Task 7: Start all Directory Replication Server on related Nodes

Start the Oracle directory replication servers on the new supplier, the old supplier and the consumer, by typing:

```
oidctl connect=db_connect_string server=oidrepld instance=instance_number \
host=hostname flags='\-p port_number' start
```

See Also:
- Chapter 6, "Process Control of Oracle Internet Directory Components"
- The `oidctl` command-line tool reference in *Oracle Identity Management User Reference.*

Performing a Time-Based Replication Failover

This section explains how to perform a time-based replication failover. It contains these topics:

- Task 1: Configure Change Log Garbage Collection Object on New Supplier
- Task 2: Save Last Change Number from New Supplier
- Task 3: Enable Change Log Regeneration on New Supplier
- Task 4: Wait for the Desired Time Period to Elapse
- Task 5: Stop all Directory Replication Servers on Related Nodes
- Task 6: Break Old Replication Agreement and Set Up New Agreement
- Task 7: Update Last Applied Change Number of New Agreement
- Task 8: Clean Up Old Agreement on Old Supplier
- Task 9: Start All Directory Replication Servers on Related Nodes

Task 1: Configure Change Log Garbage Collection Object on New Supplier

Configure the changelog purging configuration entry on the new supplier so that it preserves change logs for the desired period of time, for example, 24 hours, as follows:

1. Create an LDIF file similar to this:

   ```
   dn: cn=changelog purgeconfig,cn=purgeconfig,cn=subconfigsubentry
   changetype:modify
   replace: orclpurgetargetage
   orclpurgetargetage: 24
   ```

2. Apply the LDIF file by typing:

   ```
   ldapmodify -p port -h host -D cn=orcladmin -w admin_pwd -f LDIF_file
   ```

   See Also: Chapter 26, "Garbage Collection in Oracle Internet Directory"

Task 2: Save Last Change Number from New Supplier

Obtain the last change number from the new supplier, as follows:

```
ldapsearch -h new_supplier_host -p port_number -D cn=orcladmin -w admin_pwd \  
-b ** -s base "objectclass=" lastchangenumber
```
Save this number!

**Task 3: Enable Change Log Regeneration on New Supplier**
Enable change log regeneration at the new supplier, as follows:

1. Create an LDIF file like this:
   ```
   dn:
   changetype: modify
   replace: orcldiprepository
   orcldiprepository: TRUE
   ```

2. Apply the LDIF file by typing:
   ```
   ldapmodify -D "cn=orcladmin" -w password -h host_name -p port_number \
   -f LDIF_file
   ```

**Task 4: Wait for the Desired Time Period to Elapse**
Wait for a period of time no greater than the value of orclpurgetargetage in the changelog purging configuration entry.

**Task 5: Stop all Directory Replication Servers on Related Nodes**
Stop the Oracle directory replication servers on the new supplier, old supplier and consumer, by typing:

```
oidctl connect=db_connect_string server=oidrepld instance=instance_number
host=hostname stop
```

**See Also:**
- Chapter 6, "Process Control of Oracle Internet Directory Components"
- The `oidctl` command-line tool reference in Oracle Identity Management User Reference.

**Task 6: Break Old Replication Agreement and Set Up New Agreement**
Break the old replication agreement between the old supplier and the consumer, then set up a new agreement between the new supplier and the consumer. Do this by using the Replication Environment Management Tool, as follows:

```
remtool -pchgmaster [-v] [-bind hostname:port_number/replication_dn_password]
```

**See Also:** The `remtool` command-line tool reference in Oracle Identity Management User Reference.

**Task 7: Update Last Applied Change Number of New Agreement**
Modify the new agreement at the new supplier so that its last applied change number has the value you retrieved in "Task 2: Save Last Change Number from New Supplier", as follows:

1. Create an LDIF file with the retrieved last applied change number, similar to this:
   ```
   dn: agreement_dn
   changetype: modify
   replace: orclLastAppliedChangeNumber
   orclLastAppliedChangeNumber: last_change_number_retrieved
   ```
2. Apply the LDIF file to the agreement by using `ldapmodify`:

```
ldapmodify -D "cn=orcladmin" -w password -h host_name -p port_number \ -f LDIF_file
```

**Task 8: Clean Up Old Agreement on Old Supplier**

If the old supplier was down when you performed "Task 6: Break Old Replication Agreement and Set Up New Agreement", the old agreement on the old supplier was not cleaned up. Clean it up now by using the Replication Environment Management Tool. Type:

```
remtool -pcleanup -agrmt [-v] [-bind hostname:port_number/replication_dn_password]
```

**See Also:** The `remtool` command-line tool reference in *Oracle Identity Management User Reference*

**Task 9: Start All Directory Replication Servers on Related Nodes**

Start the Oracle directory replication servers on the new supplier, the old supplier and the consumer, by typing:

```
oidctl connect=db_connect_string server=oidrepld instance=instance_number host=hostname start
```

**See Also:**

- Chapter 6, "Process Control of Oracle Internet Directory Components"
- The `oidctl` command-line tool reference in *Oracle Identity Management User Reference*. 
This chapter tells you how to monitor and manage replication in Oracle Internet Directory.

Once you have installed and configured replication, you can view or modify the default values for replication-related objects. This section contains these topics:

- Viewing and Modifying Directory Replication Server Configuration Parameters
- Viewing and Modifying Parameters for Particular Replica Nodes
- Modifying Parameters for Replication Agreements
- Changing the Replication Administrator's Password on All Nodes Using Oracle Database Advanced Replication
- Managing the Change Log
- Modifying the Speed of Directory Replication
- Managing and Monitoring Topology
- The Compare and Reconcile Tool

See Also:

- "The Replication Agreement Entry" on page 29-10
- "The Replica Subentry" on page 29-9

Note: No change to any configuration parameter or replication agreement takes effect until the replication server is restarted.

## Viewing and Modifying Directory Replication Server Configuration Parameters

The section "Directory Server Configuration Parameters" under "Replication Schema Elements" in *Oracle Identity Management User Reference* lists and describes the directory replication server configuration parameters. These parameters are stored in the replication server "configuration set entry", which has the following DN: `cn=configset0,cn=osdrepld,cn=subconfigsubentry`. This entry contains replication attributes that control replication processing. You can modify some of these attributes.
Viewing Configuration Parameters of the Directory Replication Server by Using Oracle Directory Manager

To view configuration parameters of the directory replication server:

1. In the navigator pane, expand in succession Oracle Internet Directory Servers, directory server instance, Server Management.
2. Select Replication Server. The following tab pages appear in the right pane.
   - Active Replication Servers, which tells you which directory replication servers are now running
   - Replication Status, which tells you the number of the last change applied from each supplier to each consumer in the DRG
   - Changelog Subscriber Status, which lists subscribers to the change log, and gives the number of the last change applied from this node

Modifying Configuration Parameters of the Directory Replication Server by Using Oracle Directory Manager

To modify configuration parameters of the directory replication server:

1. In the navigator pane, expand Oracle Internet Directory Servers, directory server instance, Server Management, Replication Server.
2. Select the replication configuration set whose parameters you want to modify. The corresponding tab pages appear in the right pane.
3. In the General tab page, modify the fields as appropriate.
   For a description of these fields, see Table A–20, "Fields in the Replication Server Configuration Set: General Tab Page".
4. For Advanced Replication-based agreements, in the ASR Agreement tab page, modify the fields as appropriate.
   For a description of these fields, see Table A–21, "Fields in the ASR Agreement Tab Page".
5. Restart the directory replication server to effect your changes.

**Note:** Be sure to add all host names for all nodes in the DRG into the Replication Group Nodes field. Do this for all nodes in the DRG.

Modifying Directory Replication Server Configuration Parameters by Using Command-Line Tools

To modify replication configuration parameters by using command-line tools, use the syntax documented in the ldapmodify command-line tool reference in Oracle Identity Management User Reference.

"Replication Schema Elements" in Oracle Identity Management User Reference describes the replication server configuration parameters. As noted in that table, the modifiable replication configuration parameters are:

- orclChangeRetryCount
- orclThreadsPerSupplier
The `orclThreadsPerSupplier` configuration parameter has two subtypes:

- **apply**: The number of worker threads for applying change logs
- **transport**: The number of worker threads for transporting change logs

**Example 31–1  Modifying the Number of Retries Before a Change Is Moved into the Purge Queue**

This example uses an input file named `mod.ldif` to change the number of retry attempts from the default of ten times to five times. Specifically, after attempting to apply an update five times, the update is dropped and logged in the replication log.

1. Edit the example file `mod.ldif` as follows:
   ```
   dn: cn=configset0,cn=osdrepld,cn=subconfigsubentry
   changetype: modify
   replace: orclChangeRetryCount
   orclChangeRetryCount: 5
   ```

2. Use `ldapmodify` to update the replication server `configset0` parameter value as follows:
   ```
   ldapmodify -D "cn=orcladmin" -w administrator_password -h my_host -p port -f mod.ldif
   ```

3. Restart the directory replication server.

**Example 31–2  Modifying the Number of Worker Threads Used for Applying Change Logs**

This example uses an input file named `mod.ldif` to change the number of worker threads used for applying change logs to 5.

1. Edit the example file `mod.ldif` as follows:
   ```
   dn: cn=configset0, cn=osdrepld, cn=subconfigsubentry
   changetype: modify
   replace: orclthreadspersupplier;apply
   orclthreadspersupplier;apply: 5
   ```

2. Use `ldapmodify` to update the replication server `configset0` parameter value as follows:
   ```
   ldapmodify -h consumer_host_name -p consumer_port -D cn=orcladmin -w administrator_password -v -f mod.ldif
   ```

3. Restart the directory replication server:
   ```
   oidctl connect=connect_string server=oidrepld instance=instance_number restart
   ```

See Also: The `oidctl` command-line tool reference in *Oracle Identity Management User Reference* for instructions on restarting the directory replication server

**Example 31–3  Modifying the Number of Worker Threads Used for Transporting and Applying Change Logs**

This example uses an input file named `mod.ldif` to make the following changes:

- Change the number of worker threads used for applying change logs to 1
- Change the number of worker threads used for transporting change logs to 1

All steps are the same as in Example 31–2. Only the LDIF file differs.
Edit the example file `mod.ldif` as follows:

```
dn: cn=configset0, cn=osdrepld, cn=subconfigsubentry
changetype:modify
replace: orclthreadspersupplier;transport
orclthreadspersupplier;transport: 1
replace: orclthreadspersupplier;apply
orclthreadspersupplier;apply: 1
```

Once you make this change, only one worker thread and one reader thread will be spawned for each replication direction between two replicas of a replication agreement.

### Viewing and Modifying Parameters for Particular Replica Nodes

To modify a particular replica node, you modify the replica subentry. "Replication Schema Elements" in *Oracle Identity Management User Reference* describes the parameters you can modify in the replica subentry.

---

**Note:** Because the directory replication server reads replication node objects from the consumer, you must apply all changes to the consumer and, optionally, to the supplier.

---

**See Also:** "The Replica Subentry" on page 29-9 for more information about the replica subentry

### Viewing and Modifying Parameters for a Particular Replica Node by Using Oracle Directory Manager

To view and modify a particular replica node by using Oracle Directory Manager:

1. In the navigator pane, expand *Oracle Internet Directory Servers*, *directory server instance*, *Replication Management*.
2. Select the replica node you want to view or modify. The corresponding tab pages appear in the right pane.
3. In the General tab page, you can modify the fields as appropriate. Table A–22, "Fields in the Replica Node: General Tab Page" describes the fields in this tab page.
4. The Replica Agreements tab page enables you to view the details of the replication agreement in which the specified node participates. The columns in this tab page are described in Table A–23, "Columns in the Replica Agreements Tab Page".
5. After you have viewed and modified the replica node, restart the directory replication server.

### Modifying a Particular Replica Node by Using Command-Line Tools

To modify replication configuration parameters by using command-line tools, use the syntax documented in the `ldapmodify` command-line tool reference in *Oracle Identity Management User Reference*.
Example 31–4  Modifying the orclReplicaURI Attribute for a Particular Replica Node

The directory replication server uses the orclReplicaURI attribute value of the replica subentry to locate the directory server for that replica. If the port or host where the directory server is running is changed, then this attribute must be modified accordingly.

1. Edit the example file mod.ldif as follows:

   Dn: orclreplicaid=unique_replica_identifier, cn=replication configuration
   Changetype:modify
   Replace:orclReplicaURI
   OrclReplicaURI: ldap://host_name:port_number/

2. Use ldapmodify to update the replica subentry orclreplicauri attribute.

   ldapmodify -D "cn=orcladmin" -w administrator_password -h my_host \ -p port -f mod.ldif

3. Restart the directory replication server.

Example 31–5  Modifying the orclReplicaSecondaryURI Attribute for a Particular Replica

The directory replication server uses the orclReplicaSecondaryURI attribute value as an alternate location to contact the directory server for a particular replica. A user can add an alternate ldapURI attribute at which the directory server can be contacted for that particular replica. To add additional ldapURI attribute:

1. Edit the example file mod.ldif as follows:

   Dn: orclreplicaid=unique_replica_identifier, cn=replication configuration
   Changetype:modify
   add:orclReplicaSecondaryURI
   OrclReplicaSecondaryURI: ldap://host_name:port_number/

2. Use ldapmodify to update the replica subentry OrclReplicaSecondaryURI attribute.

   ldapmodify -D "cn=orcladmin" -w administrator_password -h my_host \ -p port -f mod.ldif

3. Restart the directory replication server.

Example 31–6  Modifying the orclReplicaState Attribute for a Particular Replica

OrclReplicaState represents the state of a particular replica. To bootstrap (re-initialize) a replica, update this attribute in the following manner:

1. Edit the example file mod.ldif as follows:

   Dn: orclreplicaid=unique_replicaID, cn=replication configuration
   Changetype:modify
   replace:orclReplicaState
   OrclReplicaState: 0

2. On the host where the consumer replica is running, use ldapmodify to update the replica subentry orclreplicastate attribute.

   ldapmodify -D "cn=orcladmin" -w administrator_password -h consumer_host \ -p port -f mod.ldif

3. Restart the directory replication server.
Modifying Parameters for Replication Agreements

This section contains instruction for modifying replication agreements that are based on both Advanced Replication and LDAP.

Modifying Parameters for Replication Agreements Based on Oracle Database Advanced Replication

Replication agreement parameters based on Advanced Replication are stored in replication agreement entries, which have the following DN:

orclAgreementID=000001,cn=replication configuration

---

**Note:**
- For replication agreements based on Advanced Replication, in the parameter DirectoryReplicationGroupDSAs, enter the host names for all of the nodes in the DRG. This list must be identical on all the nodes.
- For Oracle Internet Directory 10g (10.1.4.0.1), only one replication agreement based on Oracle Database Advanced Replication can be used. The DN of this replication agreement is orclagreementid=000001,cn=replication configuration.
- Before you modify replication agreement parameters, be sure that you have started the Oracle Internet Directory on all nodes.

---

See Also:
- "Viewing and Modifying Replication Agreements Based on Oracle Database Advanced Replication by Using Oracle Directory Manager" on page 31-6
- "Managing Replication Agreements Based on Advanced Replication by Using ldapmodify" on page 31-7

---

Viewing and Modifying Replication Agreements Based on Oracle Database Advanced Replication by Using Oracle Directory Manager

To view and modify replication agreement parameters by using Oracle Directory Manager:

1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance, then Replication Management. The following tab pages appear in the right pane:
   - Replication Status, which tells you the number of the last change applied from each supplier to each consumer in the DRG
   - Replica Status, which tells you the state of the replica—that is, whether it is online, offline, or in the bootstrapping process.
   - Changelog Subscriber, which lists subscribers to the change log, and gives the number of the last change applied from this node
ASR Agreement, in which you can view and modify the information for an Advanced Replication-based replication agreement. The fields in this tab page are described in Table A–21, "Fields in the ASR Agreement Tab Page".

Note: Be sure to add all host names for all nodes in the DRG into the Replication Group Nodes field. Do this for all nodes in the DRG.

2. If you are modifying the values, and want to return to the values that appeared when you first opened this pane, then click Revert. If you are satisfied with your changes, then click Apply.

Managing Replication Agreements Based on Advanced Replication by Using ldapmodify

"Replication Schema Elements" in Oracle Identity Management User Reference lists and describes the replication agreement parameters and indicates those that you can modify.

To add more nodes to the values in a replication agreement entry, run ldapmodify at the command line, referencing an LDIF-formatted file.

Example 31–7 Adding Nodes to a Replication Agreement

This example uses an input file named mod.ldif to add two nodes to a replication agreement:

1. Edit mod.ldif as follows:

   dn: orclagreementid=000001,cn=replication configuration
   changetype: modify
   add: orcldirreplgroupdsas
   orcldirreplgroupdsas: hollis
   orcldirreplgroupdsas: eastsun-11

2. Use ldapmodify to update the replication server configset0 parameter value as follows:

   ldapmodify -D "cn=orcladmin" -w administrator_password -h host \
   -p port -f mod.ldif

3. Restart the directory replication server.

This procedure modifies the entry containing the replication agreement whose DN is orclagreementid=000001,cn=replication configuration. The input file adds the two nodes, hollis and eastsun-11, into the replication group governed by orclagreementid=000001.

Note: You must include the new nodes—for example, hollis and eastsun-11 in the previous sample LDIF file—in the orclDirReplGroupDSAs parameter on each node in the replicated environment before you start the replication process.

"Adding a Node for Multimaster Replication (Oracle Database Advanced Replication Types Only)" on page 30-14 explains the process of adding a new node to a replication environment.
Because Oracle Internet Directory 10g (10.1.4.0.1) supports only one configuration set for the directory replication server, you do not need to specify a configuration set.

**Example 31–8  Modifying the orclExcludedNamingContexts Attribute for an Oracle Database Advanced Replication Replica Agreement**

In a replication agreement based on Advanced Replication, the directory replication server uses the value of the orclExcludedNamingcontexts attribute of the replica agreement entry to specify the top level subtrees to be excluded from replication.

In this example, two top level naming contexts—c=us and c=uk—are excluded from Advanced Replication.

1. Edit the example file mod.ldif as follows:
   
   ```
   dn: orclAgreementID=000001, cn=replication configuration
   Changetype:modify
   Replace: orclExcludedNamingcontexts
   orclExcludedNamingcontexts: c=us
   orclExcludedNamingcontexts: c=uk
   ```
   
   2. Use ldapmodify to update the replication agreement orclupdateschedule attribute.
   
   ```
   ldapmodify -D "cn=orcladmin" -w administrator_password -h consumer_host \
   -p port -f mod.ldif
   ```
   
   3. Restart the directory replication server.

**Modifying Parameters for Replication Agreements Based on LDAP**

LDAP-based replication agreement parameters are stored in replication agreement entries, which have the following DN:

- `orclAgreementID=unique_identifier_of_the_replication_agreement`
- `orclReplicaId=unique_identifier_of_the_supplier`, `cn=replication configuration`

**Note:** Ensure that the agreement is identical at both the supplier and the consumer. The replication server reads the last applied change number and the naming context from the agreement at the supplier node. It reads the other agreement attributes from the consumer.

**Viewing and Modifying LDAP-Based Replication Agreement Parameters by Using Oracle Directory Manager**

To view and modify replication agreement parameters by using Oracle Directory Manager:

1. In the navigator pane, expand in succession Oracle Internet Directory Servers, directory server instance, Replication Management, Replica Node: replica identifier.

2. Select the replica agreement you want to view or modify. The following tab pages appear in the right pane:
   
   - **General**, in which you can view and modify LDAP based replication agreement information. The fields in this tab page are described in Table A–23, “Columns in the Replica Agreements Tab Page”.

31-8  Oracle Internet Directory Administrator’s Guide
Modifying Parameters for Replication Agreements

- **Replica Naming Context**, in which you can view, add, delete, and modify LDAP naming context objects. The fields in this tab page are described in Table A–24, "Fields in the Replica Agreement: Replica Naming Context Tab Page".

Modifying LDAP-Based Replication Agreement Parameters by Using ldapmodify

"Replication Schema Elements" in Oracle Identity Management User Reference describes the replication agreement parameters and indicates those that you can modify.

**Example 31–9  Modifying the orclUpdateSchedule Attribute for a Particular Replica Agreement**

The directory replication server uses the orclupdateschedule attribute value of the replica agreement entry as time interval in minutes to determine how often the replication server process the new change logs from the supplier.

This example shows that replication server will process new change logs from the supplier for every minute.

1. Edit the example file `mod.ldif` as follows:
   
   ```
   dn: orclAgreementID=id_number,orclReplicaId=replica_identifier,
   cn=replication configuration
   Changetype:modify
   Replace: orclupdateschedule
   orclupdateschedule: 1
   ```

2. Use ldapmodify to update the replication agreement `orclupdateschedule` attribute.
   
   ```
   ldapmodify -D "cn=orcladmin" -w administrator_password -h consumer_host \ 
   -p port -f mod.ldif
   ```

3. Restart the directory replication server.

**Example 31–10  Modifying the orclLastAppliedChangeNumber Attribute for a Particular Replica Agreement**

The directory replication server uses the value `orclLastAppliedChangeNumber` attribute to determine the number of last applied change log processed by the consumer.

The modification of the `orclLastAppliedChangeNumber` attribute must be applied against the supplier node since replication server reads `orclLastAppliedChangeNumber` from the same duplicate agreement at the supplier.

In this example, `orclLastAppliedChangeNumber` attribute of the duplication agreement at the supplier is set to 700, which indicates all change logs with changelog number prior to 700 has been processed by the replication server.

**Note:** Do not modify the `orclLastAppliedChangeNumber` attribute except as instructed during the partial replication add node procedure.

1. Edit the example file `mod.ldif` as follows:
   
   ```
   dn: orclAgreementID=unique_identifier_of_the_replication_agreement,
   orclReplicaId=unique_identifier_of_the_supplier, cn=replication configuration
   Changetype:modify
   ```
Changing the Replication Administrator's Password on All Nodes Using Oracle Database Advanced Replication

You can change the password for the replication administrator database account on all nodes of a DRG using Oracle Database Advanced Replication by using the `-chgpwd` argument to the Replication Environment Management Tool, `remtool`. To use this argument, enter:

```
remtool -chgpwd
```

The `remtool` utility then prompts you for the MDS Global Name—that is, the name of the Master Definition Site—the current password, and the new password. It then asks you to confirm the new password. If you enter an incorrect current password, then you must run the Replication Environment Management Tool again.

You can also use the `-pchgpwd` argument to `remtool` to change the password of the replication DN of a replica.

To change the password only in the replication wallet, `$ORACLE_HOME/dap/admin`, use the `-pchgwalpwd` argument to `remtool`. To use this argument, enter:

```
remtool -pchgwalpwd
```

See Also: The `remtool` command-line tool reference in *Oracle Identity Management User Reference* for more information about using this tool.

Managing the Change Log

Oracle Directory Manager enables you to view the last 25 changes you performed, listing them by change log number, the type of operation—namely, add, modify, or delete—in which each occurred, and the entry on which each was made. It allows you to select a particular change to see more specific details about it.

To manage the change log, in the navigator pane expand in succession *Oracle Internet Directory Servers*, *directory server instance*, then select *Change Log Management*. The right pane lists the last 25 changes, beginning with the most recent. It tells you the change number, the type of operation in which each change occurred, and the entry on which the change was made.

To see the details of a particular change, in the right pane, select the change, then choose *View Properties*. The Change Log window appears. The fields for the Change Log window are listed and described in Table A–27, "Fields in the Change Log Window".
Modifying the Speed of Directory Replication

In the default configuration for replication, the `orclupdateschedule` attribute is set to a value of 1, representing 1 minute. You can shorten the replication processing time by changing the value of the `orclupdateschedule` attribute to 0, representing 1 second.

Modifying the Speed of Directory Replication When Using Oracle Database Advanced Replication

In directory replication based on Advanced Replication, the default configuration achieves a processing time that is approximately 2.5 minutes:

- 1 minute for the supplier to prepare the change for sending to the consumer
- 30 seconds for Advanced Replication to push the change to the consumer
- 1 minute for the consumer to apply the change

In the case of Advanced Replication, changing the default value for the `orclupdateschedule` attribute to 0 results in a replication time of 32 seconds. To do this:

1. Edit `mod.ldif` as follows:

   ```
   dn: orclagreementid=000001, cn=replication configuration,
   cn=replication configuration
   changetype:modify
   replace: orclupdateschedule
   orclupdateschedule: 0
   ```

2. Upload `mod.ldif` as follows:

   ```
   ldapmodify -D "cn=orcladmin" -w administrator_password -h host_name -p port \
   -v -f mod.ldif
   ```

3. Restart the directory replication server:

   ```
   oidctl connect=connect_string server=oidrepld instance=instance_number restart
   ```

Modifying the Speed of Directory Replication When Using LDAP-Based Replication

In LDAP-based directory replication, the default configuration achieves a processing time that is approximately 1 minute during which the change is retrieved from the supplier and applied to the consumer. Changing the default value for the `orclupdateschedule` attribute to 0 results in a replication time of 1 second. To do this:

1. Edit `mod.ldif` as follows:

   ```
   dn: orclAgreementID=unique_identifier_of_the_replication_agreement,
   orclReplicaId=unique_identifier_of_the_supplier,
   cn=replication configuration
   changetype:modify
   replace: orclupdateschedule
   orclupdateschedule: 0
   ```

2. On the consumer host, upload `mod.ldif` as follows:

   ```
   ldapmodify -h consumer_host_name -p consumer_port -D cn=orcladmin \ 
   -w administrator_password -v -f mod.ldif
   ```
Managing and Monitoring Topology

The Replication Environment Management Tool, remtool, enables you to monitor the health of the replication process. You can run remtool periodically to ensure that your replication processes are performing properly.

The remtool options for monitoring the health of the replication process are -pdipqstat and -pverify options to remtool. These are also known as the Display Queue Statistics Tool and the Replication Verification Tool. Their syntax is as follows:

```
remtool -pdipqstat [-v] [-bind hostname:port_number/replication_dn_password]
remtool -pverify [-v] [-bind hostname:port_number/replication_dn_password]
  [-hiqmax hiqmax] [-tbtmax tbtmax]
```

First run the Display Queue Statistics Tool. It will show the queue statistics of the DRG. Check to see if the number of Human Intervention Queue (HIQ) entries and change logs to be transported (Logs TBP) are higher than usual. If so, that means replication is running more slowly than it should. Run the Replication Verification Tool to verify your replication configurations.

If the Replication Verification Tool reports test failure, check the report that it generates and follow the suggestion in the report to fix the specific failures.

The Compare and Reconcile Tool

The compare and reconcile tool oidcmprec enables you to compare two directories. It detects and resolves conflicts. Of the two directories, one directory is considered to be the source directory or "source of truth." The other directory is the destination directory that needs to be synchronized with the source directory. The directories to be compared can be standalone directories, part of the same replication group, or part of different replication groups.

This section provides an introduction to the oidcmprec tool and some examples of oidcmprec usage. It includes the following topics:

- Conflict Scenarios
- Operations Supported by oidcmprec
- Output from oidcmprec
- How oidcmprec Works
- Setting the Source and Destination Directories
- Selecting the DIT for the Operation
- Selecting the Attributes for the Operation
- Controlling Change Log Generation
- Using a Parameter File
- Including Directory Schema
- Overriding Predefined Conflict Resolution Rules
- Using the User-Defined Compare and Reconcile Operation
Known Limitations of the oidcmprec Tool

See Also: The "oidcmprec" command reference in Oracle Identity Management User Reference for the complete syntax for oidcmprec, including operations, conflict scenarios, and conflict resolution rules.

Conflict Scenarios

The oidcmprec tool can detect and resolve the following conflict scenarios:

- Entry only in source directory (entos)
- Entry only in destination directory (entod)
- Attribute only in source directory (atros)
- Attribute only in destination directory (atrod)
- Single-valued attribute differs (svatrdif)
- Multi-valued attribute differs (mvatrdif)
- Entry DN differs (dndif)

The dndif scenario can occur in a replication environment when a modrdn or moddn operation performed in one node is not replicated to another node. As a result, the entry has the same orclguid but different DNs on the two nodes. The tool uses the orclguid attribute to detect this conflict.

The oidcmprec tool can also detect and resolve the following schema conflict scenarios:

- Object class definition exists only in source directory (odefos)
- Object class definition exists only in destination directory (odefod)
- Object class definition different in source and destination directory (odefdif)
- Attribute definition exists only in source directory (adefos)
- Attribute definition exists only in destination directory (adefod)
- Attribute definition different in source and destination directory (adefdif)

Operations Supported by oidcmprec

The tool supports five operation. Each operation compares entries, detects conflicts, and optionally resolves them. The operations differ as to how they resolve conflicts. The operations are as follows:

- Compare operation: compares two directories and stores the changes as LDIF records in a file. The LDIF file can be applied to the destination directory to make it identical to the source directory. Only the data in the source directory is considered valid.

- Reconcile operation: compares two directories and applies necessary changes at the destination directory to make it identical to the source directory. All the changes made to the directory are stored as LDIF records in a file. Only the data in the source directory is considered valid.

- Merge or two-way reconcile operation: compares two directories and applies necessary changes at the source or destination directory to make them identical. The data in both directories is considered valid. For example, when the tool detects that an entry exists only in the destination directory, the tool adds it to the
source. This operation also records all the changes applied to the directory as LDIF records in a file.

- **Merge dry run operation**: compares two directories, similar to the merge operation, but does not apply the changes in the directory. Instead, it stores the changes as LDIF records in a file. The changes to be applied to the source directory and to destination directory are stored in two different files.

- **User defined compare and reconcile operation**: uses conflict resolution rules that you choose for each conflict scenario. See Oracle Identity Management User Reference for a list of conflict resolution rules you can use for each conflict scenario.

### Output from oidcmprec

The `oidcmprec` tool generates the following files for every operation:

- `filename.rpt`: contains the DNs of all entries compared and the compare result.
- `filename.s2d.ldif`: contains all changes applied to the destination directory or stored for later application to the destination directory. The name is an abbreviation for source directory to destination directory.
- `filename.d2s.ldif`: contains all changes applied to the source directory or stored for later application to the source directory. The name is an abbreviation for destination directory to source directory.
- `filename.eos.rpt`: lists DNs of entries that exist only in the source directory. It also lists name of attributes and object classes that are defined only in the source directory, if the schema is included for the operation. The name is an abbreviation for entries available only in source directory.
- `filename.eod.rpt`: lists DNs of entries that exist only in the destination directory. It also lists name of attributes and object classes that are defined only in the destination directory, if the schema is included for the operation. The name is an abbreviation for entries available only in destination directory.
- `filename.dif.rpt`: lists DNs of all entries that differ along with names of attribute that differ. It also lists name of attributes and object classes whose definitions differ, if the schema is included for the operation. This file is known as dif file.
- `filename.err`: contains all error messages. This file is known as err file.

### How oidcmprec Works

Using the replication DN and replication DN password as credentials, the tool performs three tasks. First it loads schema information into memory. Next, it collects the entries to be compared and it compares them, attribute by attribute. The tool uses schema information to determine the compare rule to be used for each attribute. Then, based on the compare result, the tool takes the necessary actions. These operations are performed by different threads:

- **The DN thread** is responsible for collecting entries to be compared. While collecting entries, it does not fetch the entire tree at once. It first fetches the base entry and processes it. Then it fetches the immediate children of the base entry and processes them, and then their immediate children, and so on. The collected entries are passed on to worker threads. You can control the number of DN threads using the `dnthreads` argument.

- **The worker thread** is responsible for comparing entries attribute by attribute and applying conflict resolution rules. The worker thread then passes on the entry to
The log writer thread. You can control the number of worker threads by using the threads argument. The total number of worker threads and DN threads cannot exceed a maximum value equal to 6 * (Number of CPUs) - 2. If you specify more than that, the tool adjusts the number of worker and DN threads so as not to exceed the maximum.

- The log writer thread is responsible for writing the contents to all seven output files listed in "Output from oidcmprec" on page 31-14. There is only one log writer thread. You cannot increase the number.

These threads are spawned, monitored and terminated by the main thread. The main thread processes command line arguments and parameter files and spawns the other threads. As soon as the main thread detects that all operations are complete, it terminates all threads and cleans up all connections.

Each thread establishes an LDAP connection to the source directory and to the destination directory. These connections remain open until all operations are completed by the thread. If a connection is closed for any reason, the tool will try to reestablish connection if the contonerr argument is set to TRUE. If the tool can reestablish the connection, it continues operation.

---

**Note:** Use the contonerr argument to specify whether the tool should continue processing on error. This argument can be set to TRUE or FALSE. By default it is set to TRUE.

---

### Setting the Source and Destination Directories

You use the source and destination options to set the source and destination directories.

```
oidcmprec source=staqj13:3060 destination=staqj:3070 base="''" \
    scope=subtree file=temp operation=compare
```

The tool prompts for passwords if they are not provided on the command line:

Enter replication DN password of the source directory :  
Enter replication DN password of the destination directory :

### Selecting the DIT for the Operation

Use the base, dns2exclude, and scope options to choose the area to be compared and reconciled.

This example compares the entire directory except c=us, dc=mycom, dc=com and c=uk, dc=mycom, dc=com:

```
oidcmprec base="''" \
    dns2exclude="'c=us,dc=mycom,dc=com' 'c=uk,dc=mycom,dc=com'" \
    operation=compare scope=subtree \
    source=myhost1.mycom.com:389/replication_dn_pwd \
    destination=myhost2.mycom.com:389/replication_dn_pwd \
    threads=5 dnthreads=2 file=cmpres
```

This example compares the naming contexts dc=com and dc=org except for the trees c=us, dc=mycom, dc=com and c=uk, dc=myorg, dc=org.

```
oidcmprec base="'dc=com' 'dc=org'" \
    dns2exclude="'c=us,dc=mycom,dc=com' 'c=uk,dc=myorg,dc=org'" \
    operation=compare scope=subtree \
    source=myhost1.mycom.com:389/replication_dn_pwd \
```
destination=myhost2.mycom.com:389/replication_dn_pwd \
threads=5 dnthreads=2 file=cmpres

Selecting the Attributes for the Operation

By default, oidcmprec compares all attributes except for the operational attributes creatorsname, createtimestamp, modifiersname, modifytimestamp, orclentrydn, and orclnormdn. You can control the attributes to be included or excluded for the chosen operation using exclattr or inclattr, respectively. The exclattr and inclattr arguments allow limited pattern matching. You can use attributename* to match all attributes starting with attributename. You can also use attributename;* to match all subtypes of attributename.

This example excludes the authpassword attribute with and without subtype, plus the attributes userpassword and category, in addition to the standard excluded attributes.

oidcmprec operation=compare scope=subtree base="'dc=com' 'dc=org'" \
source=myhost1.mycom.com:389/replication_dn_pwd \
destination=myhost2.mycom.com:389/replication_dn_pwd \
exclattr="authpassword authpassword;* userpassword category" \
threads=5 dnthreads=2 file=compare

This example includes only the attributes uid, cn, sn, givenname, and mail in the compare operation.

oidcmprec operation=compare scope=subtree base="'dc=com'" \
source=myhost1.mycom.com:389/replication_dn_pwd \
destination=myhost2.mycom.com:389/replication_dn_pwd \
inclattr="uid cn sn givenname mail" \
file=compare

This example includes all attributes for the compare operation except orclguid, creatorsname, and modifiersname. This example also asks the tool to stop when it encounters any error by setting contonerr=false.

oidcmprec operation=compare scope=subtree base="'dc=com'" \
source=myhost1.mycom.com:389/replication_dn_pwd \
destination=myhost2.mycom.com:389/replication_dn_pwd \
inclattr="*" exclattr="orclguid creatorsname modifiersname" \
file=compare contonerr=false

Controlling Change Log Generation

Change log generation for the changes made by oidcmprec depends on the value of the orcldirepository attribute of the root DSE. Change log generation behavior, however, can be controlled by using the genchglog argument. The genchglog argument can have the following values:

■ default: The directory server settings determine whether a change log is generated or not. Change logs are generated if the root entry’s orcldirepository attribute is set to true. They are not generated if orcldirepository is set to false. The same rule applies for both the source and destination directories. default is the default value for genchglog.

■ true: Change logs are always generated, irrespective of the settings on the source and destination directories.

■ false: Change logs are never generated, irrespective of the settings on the source and destination directories.
In the following example, genchglog=false to turns off change log generation:

```
oidcmprec operation=merge scope=subtree base="'dc=com'" \
    source=myhost1.mycom.com:389/replication_dn_pwd \
    destination=myhost2.mycom.com:389/replication_dn_pwd \
    inclattr="*" exclattr="orclguid creatorsname modifiersname" \
    file=merge genchglog=false
```

Using a Parameter File

All the arguments that can be given on the `oidcmprec` command line can also be stored in a parameter file. You can input the parameter file using the `paramfile` option. If you specify an argument both on the command line and in the parameter file, the argument specified on the command line takes precedence over the one specified in the parameter file. For example:

```
oidcmprec paramfile=comp_param threads=4
```

This example uses the following sample parameter file:

```
#Parameter file for compare and reconcile tool
#Creator   : John
#Date      : 21-Mar-2006
#File Name : comp_param

operation=compare
source=staqj13:3060/ods
destination=staqj13:3070/ods
base="cn=oraclecontext"
base="c=uk,dc=mycom,dc=com"
base="c=us,dc=mycom,dc=com"
verbose=false
force=true
threads=6
dnthreads=2
exclattr="orclguid userpassword authpassword authpassword;*
filename=cmp2006Feb01
```

In this example, the tool spawns four worker threads. It gives precedence to command line arguments.

Including Directory Schema

You can include the schema in an `oidcmprec` operation by including `cn=subschemasubentry` in the base argument. For example:

```
oidcmprec operation=merge scope=subtree \
    base="'dc=com' 'cn=subschemasubentry'" \
    source=myhost1.mycom.com:389/replication_dn_pwd \
    destination=myhost2.mycom.com:389/replication_dn_pwd \
    inclattr="*" exclattr="orclguid creatorsname modifiersname" \
    file=merge genchglog=false
```

If you include other DNs in addition to the schema, `oidcmprec` performs the operation on the schema first.
Overriding Predefined Conflict Resolution Rules

Conflict scenarios and conflict resolution rules for each operation are described in the "oidcmprec" command reference in Oracle Identity Management User Reference.

You can override the predefined conflict resolution rules by specifying the conflict name and the rule to use in the command line or parameter file. The following example changes the conflict resolution rule used for the conflicts dndif and mvatrdif to ignore for the compare operation:

```
oidcmprec operation=compare source=host1:3060 destination=host2:3070 \
  base="" scope=subtree file=temp operation=compare \ 
  dndif=ignore mvatrdif=ignore
```

Using the User-Defined Compare and Reconcile Operation

In addition to the predefined operations compare, reconcile, merge, and merge dry run, oidcmprec has a user-defined compare and reconcile operation, userdefinedcr, that enables you to specifying conflict resolution rule arguments. Any conflict resolution rule you do not specify with userdefinedcr defaults to ignore. The following command line uses the userdefinedcr operation:

```
oidcmprec operation=userdefinedcr scope=subtree \
  base="" scope=subtree file=temp operation=userdefinedcr \ 
  source=host1:3060/replication_dnp password=mypwd \ 
  destination=host2:3070/replication_dnp password=mypwd \ 
  entos=add entod=ignore atros=add atrod=ignore \ 
  svatrdif=usesrc mvatrdif=usesrc dndif=ignore \ 
  threads=5 dnthreads=2 file=myreconcile
```

Conflict scenarios and conflict resolution rules are described in the "oidcmprec" command reference in Oracle Identity Management User Reference.

Known Limitations of the oidcmprec Tool

The oidcmprec tool has the following limitations:

- When the tool logs changes to LDIF records in the filename.s2d.ldif or filename.d2s.ldif file for deletion of a tree, it logs the parent record first, followed by its children. If you attempt to apply this change using the ldapmodify command-line tool, it will fail, as the directory server will not allow deletion of non-leaf entry. To prevent ldapmodify from failing, edit the file to reorder the records before running ldapmodify.

- When the tool performs a delete operation on a entry, it deletes that entry and its children. The tool records that the entry was deleted, but does not log that its children were also deleted.

- The tool has limitations with respect to compound RDNs. These are RDNs that contain two or more attribute=attrvalue pairs, separated by a+, for example:

  - uid=jpaul + cn=john paul + mail=john.paul@oracle.com,dc=oracle,dc=com

If one of the directories you are comparing contains a compound RDN, when the tool suggests modrdn/moddn changes in the filename.s2d.ldif or filename.d2s.ldif file, the deleteoldrdn value might be incorrect.
Part VI

Directory Plug-ins

This part contains these chapters:

- Chapter 32, "Oracle Internet Directory Server Plug-in Framework"
- Chapter 33, "Oracle Internet Directory Plug-In for Password Policies"
- Chapter 34, "Setting Up the Customized External Authentication Plug-in"
This chapter describes how you can extend the capabilities of the Oracle directory server by using plug-ins developed by either Oracle Corporation or third-party vendors. As of 10g (10.1.4.0.1), Oracle Internet Directory supports plug-ins in Java as well as in PL/SQL.

This chapter contains these topics:

- About Directory Server Plug-ins
- Creating Plug-ins
- Registering and Managing Plug-ins

**See Also:** The chapter on the Oracle Internet Directory server plug-in framework in *Oracle Identity Management Application Developer’s Guide*.

### About Directory Server Plug-ins

Directory server plug-ins can provide the directory server with the following kinds of added functionality, to mention just a few:

- Validate data before the directory server performs an operation on it
- Perform specified actions after the server performs an operation
- Define password policies
- Authenticate users through external credential stores

On startup, the directory server loads your plug-in configuration and library. Then, when it processes requests, it calls your plug-in functions whenever the specified event takes place.

In Figure 32–1, LDAP clients, each using a separate application, send information to and receive it from the Oracle directory server. Plug-in configuration tools likewise send information to the directory server. The directory server sends data to Plug-in Module 1, Plug-in Module 2, and Plug-in Module 3. Each plug-in module has both a plug-in module interface and plug-in logic. Each plug-in module sends information to and receives it from the LDAP API and the Plug-in LDAP.
The work that plug-ins perform depends on whether they execute before, after, or in addition to normal directory server operations. The next section explains the various kinds of operation-based plug-ins.

**LDAP Operations and Timings Supported by the Directory**

The Oracle Internet Directory server supports plug-ins for the following LDAP operations:

- ldapadd
- ldapbind
- ldapcompare
- ldapdelete
- ldapmoddn (Java only)
- ldapmodify
-ldapsearch

Oracle Internet Directory supports four operation timings for plug-ins:

- pre
- post
- when
- when_replace

These are explained in the next four sections.
Pre-Operation Server Plug-ins

The server calls pre-operation plug-in modules before performing the LDAP operation. The main purpose of this type of plug-in is to validate data before the data is used in the LDAP operation.

When an exception occurs in the pre-operation plug-in, one of the following occurs:

- When the return error code indicates warning status, the associated LDAP request proceeds.
- When the return code indicates failure status, the request does not proceed.

If the associated LDAP request fails later on, the directory does not roll back the committed code in the plug-in modules.

Post-Operation Server Plug-ins

The Oracle Internet Directory server calls post-operation plug-in modules after performing an LDAP operation. The main purpose of this type of plug-in is to invoke a function after a particular LDAP operation is executed. For example, logging and notification are post-operation plug-in functions.

When an exception occurs in the post-operation plug-in, the associated LDAP operation is not rolled back.

If the associated LDAP request fails, the post plug-in is still executed.

When-Operation Server Plug-ins

The directory calls when-operation plug-in modules while performing standard LDAP operations. A when-operation plug-in executes immediately before the server’s own code for the operation. The main purpose of this type of plug-in is to augment existing operations within the same LDAP transaction. If the when-operation plug-in fails, the standard LDAP operation does not execute. If the when-operation plug-in completes successfully, but the standard LDAP operation fails, then the changes made in the plug-in are not rolled back.

You can, for example, use a when-operation plug-in with the ldapcompare operation. The directory executes its server compare code and executes the plug-in module defined by the plug-in developer.

PL/SQL when-operation plug-ins are supported in ldapadd, ldapdelete, and ldapmodify. Java when_operation plug-ins are supported in ldapadd, ldapdelete, ldapmoddn, ldapmodify, and ldapsearch.

When_Replace-Operation Server Plug-ins

A when_replace-operation plug-in executes instead of the server’s code for the operation. You can, for example, use a when_replace plug-in with the ldapcompare operation. The directory does not execute its compare code. Instead it relies on the plug-in module to perform the comparison.

PL/SQL when_replace-operation plug-ins are supported only in ldapadd, ldapcompare, ldapdelete, ldapmodify, and ldapbind.

Java when_replace-operation plug-ins are supported in ldapadd, ldapbind, ldapcompare, ldapdelete, ldapmoddn, ldapmodify and ldapsearch.
Creating Plug-ins

The plug-in framework is the environment in which you develop, configure, and apply the plug-ins. Each individual plug-in instance is called a plug-in module.

The plug-in framework includes the following:

- Plug-in configuration tools
- Plug-in module interface
- Plug-in LDAP APIs:
  - PL/SQL package `ODS.LDAP_PLUGIN`
  - Java package `oracle.ldap.ospf`

For both languages, you follow these general steps to use the server plug-in framework:

1. Write a user-defined plug-in procedure in PL/SQL or Java.
2. Compile the plug-in module.
3. Register the plug-in module through the configuration entry interface by using either the command line or Oracle Directory Manager.

Creating a PL/SQL plug-in module is like creating a PL/SQL package. Both have a specification part and a body part. The directory, not the plug-in, defines the plug-in specification because the specification serves as the interface between Oracle Internet Directory and the custom plug-in.

For security reasons and for the integrity of the LDAP server, you can compile PL/SQL plug-ins only in the ODS database schema. You must compile them in the database that serves as the backend database of Oracle Internet Directory.

Before you compile a Java plug-in, ensure that your CLASSPATH is set to `$ORACLE_HOME/ldap/jlib/ospf.jar`.

See Also: `Oracle Identity Management Application Developer’s Guide`
for more information.

Registering and Managing Plug-ins

To enable the directory server to call a plug-in at the right moment, you must register the plug-in with the directory server. Do this by creating a configuration entry for the plug-in under `cn=plugin,cn=subconfigsubentry`. This plug-in must have `orclPluginConfig` as one of its object classes.

See Also:
- The chapters on server plug-ins in `Oracle Identity Management Application Developer’s Guide`
- "Plug-in Schema Elements" in `Oracle Identity Management User Reference`

for information about the attributes in the orclPluginConfig object class

This section contains these topics:

- Registering and Managing Plug-ins by Using Oracle Directory Manager
Registering and Managing Plug-ins by Using Command-Line Tools

Registering and Managing Plug-ins by Using Oracle Directory Manager

This section provides examples of how to create, modify, and delete plug-in configuration entries by using Oracle Directory manager.

Adding a Plug-in Configuration Entry by Using Oracle Directory Manager

To register a plug-in:

1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance.
2. Select Plug-in Management. The Plug-in Management window appears in the right pane.
4. In the New Plug-in dialog box, Mandatory Properties tab page and Optional Properties tab page, enter values in the fields. These fields are described in Table A–15 on page A-9 and Table A–16 on page A-10.
5. When you have finished entering the values, choose OK. This returns you to the Plug-in Management window. The plug-in you just created is listed in the Plug-in Entry Name column.
6. Choose OK.

Editing a Plug-in by Using Oracle Directory Manager

To edit a plug-in entry:

1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance.
2. Select Plug-in Management. The Plug-in Management window appears in the right pane.
3. In the right pane, select the name of the plug-in entry you want to edit, then choose Edit. The Plug-in: dialog box appears.
4. In the Plug-in: dialog box, Mandatory Properties tab page and Optional Properties tab page, modify the values in the appropriate fields. These fields are described in Table A–17 on page A-10 and Table A–18 on page A-11. To add attributes not shown on the Mandatory Properties tab page or Optional Properties tab page, use the Advanced tab page, described in Table A–19 on page A-12.
5. Choose OK.

Deleting a Plug-in by Using Oracle Directory Manager

To delete a plug-in:

1. In the navigator pane, expand Oracle Internet Directory Servers, then directory server instance.
2. Select Plug-in Management. The Plug-in Management window appears in the right pane.
3. In the right pane, select the name of the plug-in you want to delete, then choose Edit. The Plug-in: dialog box appears.
4. In the Plug-in dialog box, choose **Delete**, and, when prompted, confirm your deletion. This returns you to the Plug-in Management window. The plug-in entry you deleted no longer appears in the list.

**Registering and Managing Plug-ins by Using Command-Line Tools**

This section provides examples of how to create, modify, and delete plug-in configuration entries by using command-line tools.

**See Also:**

- The chapters on server plug-ins in *Oracle Identity Management Application Developer’s Guide*
- "Plug-in Schema Elements" in *Oracle Identity Management User Reference*

for information about the attributes in the orclPluginConfig object class

**Example: Adding a Plug-in Configuration Entry by Using Command-Line Tools**

To add a plug-in configuration entry from the command line, create an LDIF file containing the plug-in configuration. Specify a DN under `cn=plugin,cn=subconfigsubentry`.

The following two-part LDIF file, `my_ldif_file.ldif`, creates an entry for an operation-based plug-in called `my_plugin1`:

```
 dn: cn=when_comp,cn=plugin,cn=subconfigsubentry
 objectclass: orclPluginConfig
 objectclass: top
 orclPluginName: my_plugin1
 orclPluginType: operational
 orclPluginTiming: when
 orclPluginLDAPOperation: ldapcompare
 orclPluginEnable: 1
 orclPluginVersion: 1.0.1
 orclPluginIsReplace: 1
 cn: when_comp
 orclPluginKind: PLSQL
 orclPluginUserDNList: dc=COM,c=us;dc=us,dc=oracle,dc=com;dc=org,dc=us;
 o=IMC,c=US
 orclPluginAttributeList: userpassword

dn: cn=post_mod_plugin, cn=plugin,cn=subconfigsubentry
 objectclass: orclPluginConfig
 objectclass: top
 orclPluginName: my_plugin1
 orclPluginType: operational
 orclPluginTiming: post
 orclPluginLDAPOperation: ldapmodify
 orclPluginEnable: 1
 orclPluginVersion: 1.0.1
 cn: post_mod_plugin
 orclPluginKind: PLSQL
```

Add this file to the directory with a command similar to this:

```
ldapadd -p 389 -h myhost -D binddn -w password -f my_ldif_file.ldif
```
Example: Modifying a Plug-in Configuration Entry by Using Command-Line Tools
This is an example of disabling a plug-in:

```
ldapmodify -h host_name -p port_number -D cn=orcladmin -w orcladminpwd <<EOF
dn: cn=post_mod_plugin,cn=plugin,cn=subconfigsubentry
changetype: modify
replace: orclPluginEnable
orclPluginEnable: 0
EOF
```

Example: Deleting a Plug-in Configuration Entry by Using Command-Line Tools
This is an example of deleting a plug-in:

```
ldapdelete -h host_name -p port_number -D cn=orcladmin \
- w orcladminpwd "cn=post_mod_plugin,cn=plugin,cn=subconfigsubentry"
```
Oracle Internet Directory Plug-In for Password Policies

Oracle Internet Directory uses plug-ins to add password value checking to its other password policy management capabilities. These plug-ins enable you to verify that, for example, a new or modified password has the specified minimum length. You can customize password value checking to meet your own requirements.

This chapter contains these topics:

- How the Password Policy Plug-in Works
- Example: Installing, Configuring, and Enabling a Customized Password Policy Plug-in

How the Password Policy Plug-in Works

When a user wants to add or modify a password, customized password value checking takes place as follows:

1. The client sends the directory server either an ldapadd or ldapmodify request.
2. Before the directory server makes the addition or modification, it passes the password value to the plug-in.
3. The plug-in
   a. Parses the entry
   b. Captures the userpassword attribute value in clear text
   c. Implements whatever password value checking you have specified
4. If the password meets the specification, then the plug-in notifies the directory server accordingly, and the directory server makes the addition or modification.
   Otherwise, the plug-in sends one of the following error messages to the directory server, which, in turn, passes it to the client.

   `ldap_add: UnKnown Error Encountered`
   `ldap_add: additional info: PASSWORD POLICY VIOLATION:0000X, less than 8 chars`
   `ldap_add: UnKnown Error Encountered`
   `ldap_add: additional info: PASSWORD POLICY VIOLATION:0000X, contains dictionary word`

   The same logic applies to the PRE ldapmodify plug-in.

The various kinds of value checks that a password policy plug-in could perform include:
Example: Installing, Configuring, and Enabling a Customized Password Policy Plug-in

This example uses the a PL/SQL program, `pluginpkg.sql`, which is described in "Contents of Sample PL/SQL Package pluginpkg.sql" on page 33-4. In general, this package contains:

- Two plug-in modules: `pre_add` and `pre_modify`
- One value checking function, `isGoodPwd`, which verifies that a password meets the minimum length requirement of eight characters and that it does not contain a dictionary word that is longer than four characters

Thus, in this example, if you try to add a user with the `userpassword` value less than eight characters, then the request is rejected. Similarly, if you try to modify a user password, and the new password value is less than eight characters, then the request is rejected. Also, if you try to add or modify a user with the `userpassword` `supersunday`, the password is rejected because super and sunday are dictionary words.

The dictionary is a list of words longer than four characters, initially stored in a file called `words.txt`. Before we implement the plug-in, we set up a database table and store the words into the table. To set up the table we use `create.sql`, which has the following contents:

```sql
drop table mydic;
create table mydic (word varchar2(1024));
commit;
exit;
```

Then we load the words into the table using the `sqlldr` command:

```sql
sqlldr control=words.txt userid=ods/ods_password
```

This section contains these topics:

- Loading and Registering the PL/SQL Program
- Coding the Password Policy Plug-in
- Debugging the Password Policy Plug-in
- Contents of Sample PL/SQL Package pluginpkg.sql

Loading and Registering the PL/SQL Program

Once you have implemented the standalone value checking PL/SQL program, do the following:

1. Load the plug-in package into the database. In this example, we enter:

```sql
sqlplus ods/odspwd @pluginpkg.sql
```
2. Register the plug-in. This example uses a file named pluginreg.dat, which contains the following:

```
### add plugin ###

dn: cn=pre_add_plugin,cn=plugin,cn=subconfigsubentry
objectclass:orclPluginConfig
objectclass:top
orclpluginname:pwd_plugin
orclplugintype:operational
orclplugintiming:pre
orclpluginldapoperation:ldapadd
orclpluginenable:1
orclpluginversion:1.0.1

cn:pre_add_plugin
orclpluginsubscriberdnlist:dc=com;o=IMC ,c=US

### modify plugin ###

dn: cn=pre_mod_plugin,cn=plugin,cn=subconfigsubentry
objectclass:orclPluginConfig
objectclass:top
orclpluginname:pwd_plugin
orclplugintype:operational
orclplugintiming:pre
orclpluginldapoperation:ldapmodify
orclpluginenable:1
orclpluginversion:1.0.1

cn:pre_mod_plugin
orclpluginsubscriberdnlist:dc=com;o=IMC ,c=US
orclpluginattributelist:userpassword
```

Note that, in this plug-in, we let the directory server know that there are two plug-in modules to invoke when it receives ldapadd or ldapmodify requests. We use `orclpluginsubscriberdnlist:dc=com;o=IMC,c=US` so that the plug-in is invoked ONLY if the target entry is under `dc=com` or `o=IMC,c=US`.

To add this file to the directory, enter the following:

```
ldapadd -p portnum -h hostname -D cn=orcladmin -w orcladminpwd -v \
-f pluginreg.dat
```

**Coding the Password Policy Plug-in**

You can use standard PL/SQL character functions to process the password value. Download any PL/SQL program that can do regular expression. The important thing is to integrate the value checking functions with your plug-in modules.

**Debugging the Password Policy Plug-in**

Turn on the directory server plug-in to help you examine the process and content of plug-ins.

To setup the directory server plug-in debugging, execute the following command:

```
sqlplus ods/password @$ORACLE/ldap/admin/oidspdsu.pls
```

To enable directory server plug-in debugging, execute the following command:

```
sqlplus ods/password @$ORACLE/ldap/admin/oidspdon.pls
```

To disable directory server plug-in debugging, execute the following command:
sqlplus ods/password @$ORACLE/ldap/admin/oidspof.pls

To show directory server plug-in debugging messages, execute the following command:

sqlplus ods/password @$ORACLE/ldap/admin/oidspdh.pls

To delete directory server plug-in debugging messages, execute the following command:

sqlplus ods/password @$ORACLE/ldap/admin/oidspdde.pls

**Contents of Sample PL/SQL Package pluginpkg.sql**

The script `pluginpkg.sql`, as used in this example, contains the following:

```sql
CREATE OR REPLACE PACKAGE pwd_plugin AS

PROCEDURE pre_add (ldapplugincontext IN ODS.plugincontext,
                   dn       IN VARCHAR2,
                   entry    IN ODS.entryobj,
                   rc       OUT INTEGER,
                   errmsg  OUT VARCHAR2
                   );

PROCEDURE pre_modify (ldapplugincontext IN ODS.plugincontext,
                      dn       IN VARCHAR2,
                      mods     IN ODS.modlist,
                      rc       OUT INTEGER,
                      errmsg  OUT VARCHAR2
                      );

-- Function: isGoodPwd
-- Parameter: inpwd
-- Purpose: 1. check if the password is at least
--          8 characters long
--          2. check if the password contains a
--          dictionary word (longer than 4 characters)

FUNCTION isGoodPwd(inpwd IN VARCHAR2)
RETURN INTEGER;

END pwd_plugin;
/

show error

CREATE OR REPLACE PACKAGE BODY pwd_plugin AS

FUNCTION isGoodPwd(inpwd IN VARCHAR2)
RETURN INTEGER
IS
    i NUMBER;
    ret NUMBER DEFAULT 1;
    minpwdlen NUMBER DEFAULT 8;
    len       NUMBER DEFAULT 0;
    lcount    NUMBER DEFAULT 0;
    matched   VARCHAR2(1024) DEFAULT NULL;
```
Example: Installing, Configuring, and Enabling a Customized Password Policy Plug-in

DECLARE
  CURSOR c1 IS
    SELECT word FROM mydic WHERE length(word) > 4
    AND instr(lower(inpwd), lower(word), 1, 1) > 0;
BEGIN
  plg_debug( '=== begin of ISGOODPWD ===');
  plg_debug( 'password = ' || inpwd);
  len := LENGTH(inpwd);
  plg_debug( 'password length = ' || len);

  IF len < minpwdlen THEN
    RETURN 0;
  ELSE
    OPEN c1;
    LOOP
      FETCH c1 INTO matched;
      EXIT WHEN c1%NOTFOUND;
      lcount := lcount + 1;
    END LOOP;
    plg_debug( 'count = ' || lcount);
    IF lcount > 0 THEN
      RETURN 2;
    ELSE
      RETURN ret;
    END IF;
  END IF;
  plg_debug( '=== end of ISGOODPWD ===');
EXCEPTION
  WHEN OTHERS THEN
    plg_debug( 'Exception in isGoodPwd(). Error code is ' || TO_CHAR(SQLCODE));
    plg_debug( '   ' || Sqlerrm);
    RETURN 0;
END;
PROCEDURE pre_add (ldapplugincontext IN ODS.plugincontext,
  dn IN VARCHAR2,
  entry IN ODS.entryobj,
  rc OUT INTEGER,
  errormsg OUT VARCHAR2)
IS
  inpwd VARCHAR2(256) DEFAULT NULL;
  ret   NUMBER        DEFAULT 1;
BEGIN
  plg_debug( '=== begin of PRE_ADD_PLUGIN ===');
  plg_debug( 'dn = ' || dn);

  plg_debug( 'entry obj ' || ':entryname = ' || entry.entryname);

  FOR l_counter1 IN 1..entry.attr.COUNT LOOP
    plg_debug( 'attrname[ ' || l_counter1 || ' ] = ' ||
      entry.attr(l_counter1).attrname);
    FOR l_counter2 IN 1..entry.attr(l_counter1).attrval.COUNT LOOP
      plg_debug( entry.attr(l_counter1).attrname ||
        '[ ' || l_counter1 || ' ].val[ ' || l_counter2 || ' ] = ' ||
        entry.attr(l_counter1).attrval(l_counter2));
    END LOOP;
  END LOOP;
END;

Oracle Internet Directory Plug-In for Password Policies  33-5
Example: Installing, Configuring, and Enabling a Customized Password Policy Plug-in

END LOOP;

IF entry.attr(l_counter1).attrname = 'userpassword' THEN
  inpwd := entry.attr(l_counter1).attrval(1);
  -- assuming only one attr val for userpassword
END IF;
END LOOP;

IF (inpwd IS NOT NULL) THEN
  ret := isGoodPwd(inpwd);
END IF;

IF (inpwd IS NULL OR ret = 0) THEN
  rc := 1;
  errmsg := 'PASSWORD POLICY VIOLATION:0000X, less than 8 chars';
  plg_debug( ' we got an invalid password, too short ');
ELSIF (ret = 2) THEN
  rc := 1;
  errmsg := 'PASSWORD POLICY VIOLATION:0000X, contains dictionary word';
  plg_debug( ' we got an invalid password, dictionary word ');
ELSE
  plg_debug( ' we got a good password ');
  rc := 0;
  errmsg := 'no pre_mod plugin error msg';
END IF;

plg_debug( '=== end of PRE_ADD_PLUGIN ===');

EXCEPTION
  WHEN OTHERS THEN
    plg_debug( 'Exception in PRE_ADD plugin. Error code is ' || TO_CHAR(SQLCODE));
    plg_debug( '   ' || Sqlerrm);
    rc := 1;
    errmsg := 'exception: pre_add plugin';
END;

PROCEDURE pre_modify (ldapplugincontext IN  ODS.plugincontext,
  dn       IN  VARCHAR2,
  mods     IN  ODS.modlist,
  rc       OUT INTEGER,
  errormsg OUT VARCHAR2 )
IS
  old_passwd VARCHAR2(256) DEFAULT NULL;
  new_passwd VARCHAR2(256) DEFAULT NULL;
  ret        NUMBER        DEFAULT 1;
BEGIN
  plg_debug( '=== begin of PRE_MOD_PLUGIN ===');
  plg_debug( dn);
  FOR l_counter1 IN 1..mods.COUNT LOOP
    IF (mods(l_counter1).operation = 2) AND
      (mods(l_counter1).type = 'userpassword') THEN
      FOR l_counter2 IN 1..mods(l_counter1).vals.COUNT LOOP
        new_passwd := mods(l_counter1).vals(l_counter2).val;
      END LOOP;
      IF entry.attr(l_counter1).attrname = 'userpassword' THEN
        inpwd := entry.attr(l_counter1).attrval(1);
        -- assuming only one attr val for userpassword
        END IF;
      END LOOP;
      IF (inpwd IS NOT NULL) THEN
        ret := isGoodPwd(inpwd);
      END IF;
      IF (inpwd IS NULL OR ret = 0) THEN
        rc := 1;
        errmsg := 'PASSWORD POLICY VIOLATION:0000X, less than 8 chars';
        plg_debug( ' we got an invalid password, too short ');
        ELSIF (ret = 2) THEN
          rc := 1;
          errmsg := 'PASSWORD POLICY VIOLATION:0000X, contains dictionary word';
          plg_debug( ' we got an invalid password, dictionary word ');
        ELSE
          plg_debug( ' we got a good password ');
          rc := 0;
          errmsg := 'no pre_mod plugin error msg';
        END IF;
        plg_debug( '=== end of PRE_ADD_PLUGIN ===');
      EXCEPTION
        WHEN OTHERS THEN
          plg_debug( 'Exception in PRE_ADD plugin. Error code is ' || TO_CHAR(SQLCODE));
          plg_debug( '   ' || Sqlerrm);
          rc := 1;
          errmsg := 'exception: pre_add plugin';
      END;
    END IF;
  END LOOP;
END;
END IF;

IF (mods(l_counter1).operation = 0) AND (mods(l_counter1).type = 'userpassword') THEN

FOR l_counter2 IN 1..mods(l_counter1).vals.COUNT LOOP
    new_passwd := mods(l_counter1).vals(l_counter2).val;
END LOOP;
END IF;

IF (mods(l_counter1).operation = 1) AND (mods(l_counter1).type = 'userpassword') THEN

FOR l_counter2 IN 1..mods(l_counter1).vals.COUNT LOOP
    old_passwd := mods(l_counter1).vals(l_counter2).val;
END LOOP;
END IF;

plg_debug( ' new password: ' || new_passwd);
plg_debug( ' old password: ' || old_passwd);

IF (new_passwd IS NOT NULL) THEN
    ret := isGoodPwd(new_passwd);
END IF;

IF (new_passwd IS NULL OR ret = 0) THEN
    rc := 1;
    errormsg := 'PASSWORD POLICY VIOLATION:0000X, less than 8 chars';
    plg_debug( ' we got an invalid password, too short ');
ELSIF (ret = 2) THEN
    rc := 1;
    errormsg := 'PASSWORD POLICY VIOLATION:0000X, contains dictionary word';
    plg_debug( ' we got an invalid password, dictionary word ');
ELSE
    plg_debug( ' we got a good password ');
    rc := 0;
    errormsg := 'no pre_mod plugin error msg';
END IF;

plg_debug( '=== end of PRE_MOD_PLUGIN ===');

EXCEPTION
    WHEN OTHERS THEN
        plg_debug( 'Exception in PRE_MODIFY plugin. Error code is ' || TO_CHAR(sqlcode));
        plg_debug( ' ' || sqlerrm);
        rc := 1;
        errormsg := 'exception: pre_mod plugin';
END;

END pwd_plugin;
/
show error

EXIT;
You can store user security credentials in a repository other than Oracle Internet Directory—for example, a database or another LDAP directory—and use these credentials for user authentication to Oracle components. You do not need to store the credentials in Oracle Internet Directory and then worry about keeping them synchronized. Authenticating a user by way of credentials stored in an external repository is called external authentication.

This chapter contains these topics:

- Native Authentication Contrasted with External Authentication
- Example: Installing, Configuring, and Enabling the External Authentication Plug-in

Native Authentication Contrasted with External Authentication

Authentication that relies on security credentials stored in Oracle Internet Directory is called native authentication. When a user enters her security credentials, the directory server compares them with the credentials stored in Oracle Internet Directory. If the credentials match, then the directory server authenticates the user.

Authentication that relies on security credentials stored in a directory other than Oracle Internet Directory is called external authentication. When a user enters her security credentials, the directory server compares them with the credentials stored in the other directory. This is done by using:

- A PL/SQL program that does the external authentication work
- An external authentication plug-in that invokes this PL/SQL program

Example: Installing, Configuring, and Enabling the External Authentication Plug-in

This section contains these topics:

- Sample PL/SQL Package oidexaup.sql
- Debugging the External Authentication Plug-in
- Contents of PL/SQL Package oidexaup.sql
Sample PL/SQL Package oidexaup.sql

This example uses the a PL/SQL program, oidexaup.sql, which is described in "Contents of PL/SQL Package oidexaup.sql" on page 34-4. This package is used for installing the external authentication plug-in PL/SQL package. It contains:

- Two plug-ins: namely, when_compare_replace and when_modify_replace
- One utility function: namely, get_nickname

The integrated package is the plug-in package, OIDEXTAUTH. It can also serve as a template to modify according to the requirements of your deployment.

To install, configure, and enable the external authentication plug-in, follow these steps:

1. Implement your standalone external authentication PL/SQL program. For example, if you want to authenticate users by using user names and passwords, then you should have a PL/SQL program which takes these two parameters.

   In our sample code, oidexaup.sql, auth_external is the program package name, and authenticate_user is the function that does the authentication. You need to make sure that this standalone program is working properly before you move on to next steps.

2. Integrate this standalone program into the plug-in modules.

3. Load the plug-in package into database. In this example, we enter:

   sqlplus ods/odspwd @oidexaup.sql

4. Register the plug-ins. Do this by creating and uploading an LDIF file that provides the directory server with the necessary information to invoke the plug-in.

5. This example uses a file named oidauth.ldif, which contains the following:

   ```ldif
   dn: cn=whencompare,cn=plugin,cn=subconfigsubentry
   objectclass:orclPluginConfig
   objectclass:top
   orclpluginname:oidextauth
   orclplugintype:operational
   orclplugintiming:when
   orclpluginldapoperation:ldapcompare
   orclpluginenable:1
   orclpluginversion:1.0.1
   orclPluginIsReplace:1
   cn:whencompare
   orclpluginsubscriberdnlist:dc=com;o=IMC,c=US
   orclpluginattributelist:userpassword
   orclpluginrequestgroup:$prgdn
   
   dn: cn=whenmodify,cn=plugin,cn=subconfigsubentry
   objectclass:orclPluginConfig
   objectclass:top
   orclpluginname:oidextauth
   orclplugintype:operational
   orclplugintiming:when
   orclpluginldapoperation:ldapmodify
   orclpluginenable:1
   orclpluginversion:1.0.1
   orclPluginIsReplace:1
   cn:whenmodify
   orclpluginsubscriberdnlist:dc=com;o=IMC,c=US
   orclpluginattributelist:userpassword
   orclpluginrequestgroup:$prgdn
   ```
In this file, we notify the directory server that, whenever there is an ldapcompare or ldapmodify request, there are two plug-ins to be invoked.

We use `orclpluginsubscriberdnlist:dc=com;o=IMC,c=US` so that plug-ins will ONLY be invoked if the target entry is under `dc=com` or `o=IMC,c=US`.

Replace `$prgdn` with the plug-in request group DN. This is an optional, recommended security feature. For integrating with Oracle Application Server Single Sign-On, this value is a required field. Only members of the group entered can invoke the plug-ins. You may enter multiple groups. Use a semicolon to separate entries.

The recommended defaults are:
- `cn=OracleUserSecurityAdmins,cn=Groups,cn=OracleContext`
- `cn=OracleDASAdminGroup,cn=Groups,cn=OracleContext,o=default_subscriber,dc=com`
  
  Note that the Oracle Application Server Single Sign-On server is a member of the first group. Also, be sure to replace `o=default_subscriber` with the correct value for your deployment environment.

To add this file to the directory, enter the following:

```
ldapadd -p portnum -h hostname -D cn=orcladmin -w orcladminpwd -v -f oidexauth.ldif
```

Now, everything should be ready. Use the ldapcompare command-line tool to verify that the plug-in and authentication program are working properly before you try to authenticate the user from Oracle Application Server Single Sign-On.

In our example, we also provide the plug-in code for externally modifying user password.

### Debugging the External Authentication Plug-in

Turn on directory server plug-in to help you to examine the process and content of plug-ins.

To setup directory server plug-in debugging, execute the following command:

```
sqlplus ods/password @$ORACLE_HOME/ldap/admin/oidspdsu.sql
```

To enable directory server plug-in debugging, execute the following command:

```
sqlplus ods/password @$ORACLE_HOME/ldap/admin/oidspdon.sql
```

To disable directory server plug-in debugging, execute the following command:

```
sqlplus ods/password @$ORACLE_HOME/ldap/admin/oidspdof.sql
```

To show directory server plug-in debugging messages, execute the following command:

```
sqlplus ods/password @$ORACLE_HOME/ldap/admin/oidspdsh.sql
```

To delete directory server plug-in debugging messages, please execute the following command:

```
sqlplus ods/password @$ORACLE_HOME/ldap/admin/oidspdde.sql
```
Contents of PL/SQL Package oidexaup.sql

The script oidexaup.sql, as used in this example, contains the following:

CREATE OR REPLACE PACKAGE OIDEXTAUTH AS

PROCEDURE when_compare_replace (ldapplugincontext IN ODS.plugincontext,
        result OUT INTEGER,
        dn IN VARCHAR2,
        attrname IN VARCHAR2,
        attrval IN VARCHAR2,
        rc OUT INTEGER,
        errormsg OUT VARCHAR2);

PROCEDURE when_modify_replace (ldapplugincontext IN ODS.plugincontext,
        dn IN VARCHAR2,
        mods IN ODS.modlist,
        rc OUT INTEGER,
        errormsg OUT VARCHAR2);

FUNCTION get_nickname (dn IN VARCHAR2,
        my_session IN DBMS_LDAP.session)
        RETURN VARCHAR2;

END OIDEXTAUTH;
/

SHOW ERROR

CREATE OR REPLACE PACKAGE BODY OIDEXTAUTH AS

-- We use this function to convert the dn to nickname.
-- When OID server receives the ldapcompare request, it
-- only has the dn information. We need to use DBMS_LDAP_UTL
-- package to find out the nickname attribute value of
-- the entry.

FUNCTION get_nickname (dn IN VARCHAR2,
        my_session IN DBMS_LDAP.session)
        RETURN VARCHAR2
IS
    my_pset_coll DBMS_LDAP_UTL.PROPERTY_SET_COLLECTION;
    my_property_names DBMS_LDAP.STRING_COLLECTION;
    my_property_values DBMS_LDAP.STRING_COLLECTION;
    user_handle DBMS_LDAP_UTL.HANDLE;
    user_id VARCHAR2(2000);
    user_type PLS_INTEGER;
    user_nickname VARCHAR2(256) DEFAULT NULL;
    myAttrs DBMS_LDAP.STRING_COLLECTION;
    retval PLS_INTEGER;

BEGIN
    plg_debug('=== Beginning of get_nickname() === ');
    user_type := DBMS_LDAP_UTL.TYPE_DN;
    user_id := dn;
    retval := DBMS_LDAP_UTL.create_user_handle(user_handle, user_type, user_id);
plg_debug('create_user_handle() Returns ' || To_char(retval));

retval := DBMS_LDAP_UTL.get_user_properties(my_session, user_handle, my_attrs, DBMS_LDAP_UTL.NICKNAME_PROPERTY, my_pset_coll);

plg_debug('get_user_properties() Returns ' || To_char(retval));

IF my_pset_coll.COUNT > 0 THEN
  FOR i IN my_pset_coll.first .. my_pset_coll.last LOOP
    retval := DBMS_LDAP_UTL.get_property_names(my_pset_coll(i), my_property_names);
    IF my_property_names.COUNT > 0 THEN
      FOR j IN my_property_names.first .. my_property_names.last LOOP
        retval := DBMS_LDAP_UTL.get_property_values(my_pset_coll(i), my_property_names(j), my_property_values);
        IF my_property_values.COUNT > 0 THEN
          FOR k IN my_property_values.FIRST..my_property_values.LAST LOOP
            user_nickname := my_property_values(k);
            plg_debug('user nickname = ' || user_nickname);
          END LOOP;
        END IF;
      END LOOP;
    END IF; -- IF my_property_names.count > 0
  END LOOP;
END IF; -- If my_pset_coll.count > 0

plg_debug('got user_nickname: ' || user_nickname);

-- Free my_properties
IF my_pset_coll.count > 0 then
  DBMS_LDAP_UTL.free_propertyset_collection(my_pset_coll);
END IF;

DBMS_LDAP_UTL.free_handle(user_handle);

RETURN user_nickname;
EXCEPTION
WHEN OTHERS THEN
  plg_debug('Exception in get_nickname. Error code is ' || to_char(sqlcode));
  plg_debug('   ' || Sqlerrm);
  RETURN NULL;
END;

PROCEDURE when_compare_replace (ldapplugincontext IN ODS.plugincontext, result OUT INTEGER, dn IN VARCHAR2, attrname IN VARCHAR2, attrval IN VARCHAR2, rc OUT INTEGER, errormsg OUT VARCHAR2)
IS
   retval pls_integer;
   lresult BOOLEAN;

   my_session     DBMS_LDAP.session;
   my_property_names DBMS_LDAP.STRING_COLLECTION;
   my_property_values DBMS_LDAP.STRING_COLLECTION;
   my_attrs       DBMS_LDAP.STRING_COLLECTION;
   my_pset_coll   DBMS_LDAP_UTL.PROPERTY_SET_COLLECTION;
   user_handle    DBMS_LDAP_UTL.HANDLE;

   user_id       VARCHAR2(2000);
   user_type     PLS_INTEGER;
   user_nickname VARCHAR2(60);
   remote_dn     VARCHAR2(256);

   i             PLS_INTEGER;
   j             PLS_INTEGER;
   k             PLS_INTEGER;

BEGIN
   plg_debug( '=== Begin of WHEN-COMPARE-REPLACE plug-in');
   plg_debug( 'DN = ' || dn);
   plg_debug( 'Attr = ' || attrname);
   -- plg_debug( 'Attrval = ' || attrval);
   DBMS_LDAP.USE_EXCEPTION := FALSE;
   errormsg := 'No error msg';
   rc := 0;

   -- converting dn to nickname
   my_session := LDAP_PLUGIN.init(ldapplugincontext);
   plg_debug( 'ldap_session =' || RAWTOHEX(SUBSTR(my_session,1,8)));
   retval := LDAP_PLUGIN.simple_bind_s(ldapplugincontext, my_session);
   plg_debug( 'simple_bind_res =' || TO_CHAR(retval));
   user_nickname := get_nickname(dn, my_session);
   plg_debug( 'user_nickname =' || user_nickname);

   -- unbind from the directory
   retval := DBMS_LDAP.unbind_s(my_session);
   plg_debug( 'unbind_res Returns ' || To_char(retval));

   IF (user_nickname IS NULL) THEN
      result := 32;
      errmsg := 'Can''t find the nickname';
      plg_debug( 'Can''t find the nickname');
      RETURN;
   END IF;

   plg_debug( '=== Now go to extauth ');

   BEGIN
      retval := auth_external.authenticate_user(user_nickname, attrval);
      plg_debug( 'auth_external.authenticate_user() returns = ' || 'True');
      result := 6; -- compare result is TRUE
   EXCEPTION
      WHEN OTHERS THEN
         result := 5; -- compare result is FALSE
plg_debug( 'auth_external.authenticate_user() returns = ' || 'False');
RETURN;
END;

plg_debug( '=== End of WHEN-COMPARE-REPLACE plug-in');
EXCEPTION
  WHEN OTHERS THEN
    rc := 1;
    errmsg := 'Exception: when_compare_replace plugin';
    plg_debug( 'EXCEPTION: ' || retval);
    plg_debug('Exception in when_compare. Error code is ' || to_char(sqlcode));
    plg_debug('   ' || Sqlerrm);
END;

PROCEDURE when_modify_replace (ldapplugincontext IN ODS.plugincontext,
    dn                IN  VARCHAR2,
    mods              IN  ODS.modlist,
    rc                OUT INTEGER,
    errmsg          OUT VARCHAR2
) IS
  retval pls_integer;
lresult BOOLEAN;
  my_session          DBMS_LDAP.SESSION;
  my_property_names   DBMS_LDAP.STRING_COLLECTION;
  my_property_values  DBMS_LDAP.STRING_COLLECTION;
  my_attrs            DBMS_LDAP.STRING_COLLECTION;
  my_modval           DBMS_LDAP.BERVAL_COLLECTION;
  my_pset_coll        DBMS_LDAP_UTL.PROPERTY_SET_COLLECTION;
  user_handle         DBMS_LDAP_UTL.HANDLE;
  l_mod_array         RAW(32);
  user_id             VARCHAR2(2000);
  user_type           PLS_INTEGER;
  user_nickname       VARCHAR2(2000);
  old_passwd          VARCHAR2(60) DEFAULT NULL;
  new_passwd          VARCHAR2(60) DEFAULT NULL;
  remote_dn           VARCHAR2(256);
  i                   PLS_INTEGER;
  j                   PLS_INTEGER;
  k                   PLS_INTEGER;
BEGIN
  plg_debug( '=== Begin of WHEN-MODIFY-REPLACE plug-in');
  DBMS_LDAP.USE_EXCEPTION := FALSE;
  user_type     := DBMS_LDAP_UTL.TYPE_DN;
  user_id       := dn;
  -- converting dn to nickname
  my_session := LDAP_PLUGIN.init(ldapplugincontext);
  plg_debug( 'ldap_session =' || RAWTOHEX(SUBSTR(my_session,1,8)));
  retval := LDAP_PLUGIN.simple_bind_s(ldapplugincontext, my_session);
  plg_debug( 'simple_bind_res =' || TO_CHAR(retval));
  user_nickname := get_nickname(dn, my_session);
plg_debug( 'user_nickname = ' || user_nickname);

-- unbind from the directory
retval := DBMS_LDAP.unbind_s(my_session);

FOR l_counter1 IN 1..mods.COUNT LOOP
  IF (mods(l_counter1).operation = 2) AND
     (mods(l_counter1).type = 'userpassword') THEN
    FOR l_counter2 IN 1..mods(l_counter1).vals.COUNT LOOP
      new_passwd := mods(l_counter1).vals(l_counter2).val;
    END LOOP;
  END IF;

  IF (mods(l_counter1).operation = 0) AND
     (mods(l_counter1).type = 'userpassword') THEN
    FOR l_counter2 IN 1..mods(l_counter1).vals.COUNT LOOP
      new_passwd := mods(l_counter1).vals(l_counter2).val;
    END LOOP;
  END IF;

  IF (mods(l_counter1).operation = 1) AND
     (mods(l_counter1).type = 'userpassword') THEN
    FOR l_counter2 IN 1..mods(l_counter1).vals.COUNT LOOP
      old_passwd := mods(l_counter1).vals(l_counter2).val;
    END LOOP;
  END IF;
END LOOP;

IF new_passwd IS NOT NULL AND old_passwd IS NOT NULL THEN
  BEGIN
    auth_external.change_passwd(user_nickname, old_passwd, new_passwd);
  EXCEPTION
  WHEN OTHERS THEN
    rc := 1;
    plg_debug( 'auth_external.change_passwd() raised exception.' );
    errmsg := 'auth_external.change_passwd() raised exception.' ;
    RETURN;
  END;
ELSIF new_passwd IS NOT NULL AND old_passwd IS NULL THEN
  BEGIN
    auth_external.reset_passwd(user_nickname, new_passwd);
  EXCEPTION
  WHEN OTHERS THEN
    plg_debug( 'auth_external.reset_passwd() raised exception.' );
    rc := 1;
    errmsg := 'auth_external.reset_passwd() raised exception.' ;
    RETURN;
  END;
ELSE
  rc := 1;
  errmsg := 'PLG_Exception. Not enough info to change passwd.';
END IF;
plg_debug( 'external change password succeed');
rc := 0;
errmsg := 'No when_mod_replace plguin error msg';
retval := DBMS_LDAP.unbind_s(my_session);

plg_debug( 'End of WHEN-MODIFY-REPLACE');
--COMMIT;
EXCEPTION
WHEN others THEN
  rc := 1;
  errmsg := 'PLG_Exception: when_modify_replace plugin';
  plg_debug('Exception in when_modify. Error code is ' || to_char(sqlcode));
  plg_debug('    ' || Sqlerrm);
END;
END OIDEXTAUTH;
/
SHOW ERRORS
--list

EXIT;
This part contains these appendixes:

- Appendix A, "Windows and Fields in Oracle Directory Manager"
- Appendix B, "The LDAP Filter Definition"
- Appendix D, "Globalization Support in the Directory"
- Appendix E, "Setting up Access Controls for Creation and Search Bases for Users and Groups"
- Appendix F, "The Multimaster Replication Process"
- Appendix G, "Searching the Directory for User Certificates"
- Appendix H, "LDAP Replica States"
- Appendix I, "Addition of a Directory Node by Using the Database Copy Procedure"
- Appendix J, "UNIX Authentication and User Provisioning with Oracle Internet Directory"
- Appendix K, "RFCs Supported by Oracle Internet Directory"
- Appendix L, "Troubleshooting Oracle Internet Directory"
This appendix lists and describes the various windows and fields Oracle Directory Manager. It contains these topics:

- Connection Management Fields in Oracle Directory Manager
- Access Control Management Fields in Oracle Directory Manager
- Attribute Uniqueness Fields in Oracle Directory Manager
- Garbage Collection Management Fields in Oracle Directory Manager
- Oracle Internet Directory Statistics Collection Management Fields in Oracle Directory Manager
- Password Policy Fields in Oracle Directory Manager
- Password Verifier Fields in Oracle Directory Manager
- Plug-in Management Fields in Oracle Directory Manager
- Replication Fields in Oracle Directory Manager
- Schema Management Fields in Oracle Directory Manager
- Server Management Fields in Oracle Directory Manager
- SSL Management Fields in Oracle Directory Manager
- Synchronization Fields in Oracle Directory Manager
- Server Chaining Management
Connection Management Fields in Oracle Directory Manager

Table A–1  Fields in the Credentials Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
</table>
| User       | The first time you log in, do so either as the super user or anonymously. If you intend to configure SSL features during this session, login as the super user. If you are logging in as the super user, in the User box, type cn=orcladmin. If you are logging in anonymously, leave the User box empty. If you have already set up the user's entry by using LDAP command-line tools, 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 |
| 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 Manager 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 field empty. If you want to login as a specific directory user, enter the corresponding password. See Also: "Managing Super Users, Guest Users, and Proxy Users" on page 7-8 for instructions on how to change the password |
| Server     | From the Server list, select the host containing the directory server to which you want to connect. If you are already connected to a directory server, and you want to connect to one on a different host:  
  1. Click the button to the right of the Server list. The Select Directory Servers 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. |
Table A–3 lists and describes the access control management fields in Oracle Directory Manager.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Port</td>
<td>The default port (389) appears in this field. If there is more than one directory server instance on the same host, then each directory server instance has a different port, and, when you select the directory server instance, that port number appears in this field. 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.</td>
</tr>
<tr>
<td>SSL Enabled</td>
<td>Selecting this check box causes all commands you issue by using Oracle Directory Manager to be sent over Secure Sockets Layer (SSL). You can connect to a directory server either with or without SSL. If you connect by using SSL, then Oracle Directory Manager becomes an SSL client. You can connect in this way if both of the following two conditions are met: ■ The server to which you are connecting uses SSL. If that server does not use SSL, and you select this check box, then authentication fails. ■ You have already created a wallet containing a certificate and a list of trusted certificates.</td>
</tr>
</tbody>
</table>

Table A–2 Fields in the SSL Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSL Location</td>
<td>The client wallet used in two-way authentication. If the client wallet is on the local machine, then type the wallet path and file name by using this syntax: file: absolute_path_name If the wallet is on another machine, then link to that location and enter the linked path and file name of the wallet.</td>
</tr>
<tr>
<td>SSL Password</td>
<td>The password to open the user’s wallet</td>
</tr>
<tr>
<td>SSL Authentication</td>
<td>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.</td>
</tr>
</tbody>
</table>

Access Control Management Fields in Oracle Directory Manager

Table A–3 lists and describes the access control management fields in Oracle Directory Manager.
Table A–4 lists and describes the authentication choices—that is, the methods by which users can be authenticated to the directory.

Table A–4  Fields in Authentication Choice List

<table>
<thead>
<tr>
<th>Authentication Choice</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MD5Digest.</td>
<td>Binding by using MD5Digest blocks Simple, Proxy and Anonymous access.</td>
</tr>
<tr>
<td>PKCS12</td>
<td>Binding by using PKCS12 blocks MD5Digest, Simple, Proxy and Anonymous access</td>
</tr>
<tr>
<td>Proxy</td>
<td>Binding as a proxy user. Specifying this authentication option blocks anonymous access.</td>
</tr>
<tr>
<td>Simple</td>
<td>Password-based authentication. Specifying this option blocks both Proxy and Anonymous access.</td>
</tr>
</tbody>
</table>

Table A–5 lists and describes the encryption choices—that is, the method by which data is encrypted.

Table A–5  Fields in Encryption Choice List

<table>
<thead>
<tr>
<th>Authentication Choice</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SASL</td>
<td>Simple Authentication and Security Layer</td>
</tr>
<tr>
<td>SSL No Authentication</td>
<td>Neither the client nor the server authenticates itself to the other. No certificates are sent or exchanged. In this case, SSL encryption/decryption only is used.</td>
</tr>
<tr>
<td>SSL One Way</td>
<td>Only the directory server authenticates itself to the client. The directory server sends the client a certificate verifying that the server is authentic.</td>
</tr>
</tbody>
</table>

See Also:  Bind Mode on page 18-8

Table A–6  Entities to Whom You Are Granting Access in the By Whom Tab Page

<table>
<thead>
<tr>
<th>Entity</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Everyone (*)</td>
<td>All who try to access the entry</td>
</tr>
<tr>
<td>A Specific Group</td>
<td>A previously defined group name</td>
</tr>
<tr>
<td>A Specific Entry</td>
<td>A previously defined directory entry</td>
</tr>
<tr>
<td>A Subtree</td>
<td>An entire subtree in the directory, which you select</td>
</tr>
<tr>
<td>When Session User's Distinguished Name (DN) Is Identified By Attribute</td>
<td>Anyone whose DN is an attribute in the entry. For example, you might want to grant read access to a group entry to members of the group.</td>
</tr>
</tbody>
</table>
When Session User’s Group Is Identified By Attribute
Any group whose DN is an attribute in the entry.

When Session User’s Unique ID (orclGUID) Is Identified By Attribute
The global user identifier (orclGUID) of the entry to which you want to grant or deny access for this entry.

When Session User’s Distinguished Name (DN) Matches the Accessed Entry
Anyone who has correctly logged in as the entry specified.

Table A–7 Access Rights for Attributes

<table>
<thead>
<tr>
<th>Access Right</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Read</td>
<td>Right to read attribute values. Even if read permission is available for an attribute, it cannot be returned unless there is browse permission on the entry itself.</td>
</tr>
<tr>
<td>Search</td>
<td>Right to use an attribute in a search filter</td>
</tr>
<tr>
<td>Write</td>
<td>Right to modify/add/delete the attributes of an entry.</td>
</tr>
</tbody>
</table>
| Selfwrite    | Right to add oneself to, delete oneself from, or modify one’s own entry in a list of DNs group entry attribute. Use this to allow members to maintain themselves on lists. For example, the following command allows people within a group to add or remove only their own DN from the member attribute:

```plaintext
access to attr=(member) by dnattr=(member) (selfwrite)
```

The `dnattr` selector indicates that the access applies to entities listed in the member attribute. The `selfwrite` access selector indicates that such members can add or delete only their own DN from the attribute. |
| Compare      | Right to perform compare operation on the attribute value |

Table A–8 Fields in the New Constraint Dialog Box

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute Uniqueness Constraint Name</td>
<td>Name of the attribute uniqueness constraint you are creating</td>
</tr>
<tr>
<td>Unique Attribute Name</td>
<td>The attribute you want the directory server to check</td>
</tr>
<tr>
<td>Unique Attribute Object Class</td>
<td>The object class where the attribute uniqueness constraint is enforced—for example, person. By default, it is enforced on all object classes.</td>
</tr>
</tbody>
</table>
| Unique Attribute Scope | The filter you want the directory server to use when searching for an attribute constraint. For example:

- base: Searches the root entry only
- onelevel: Searches one level only
- sub: Searches the entire directory |
| Unique Attribute Subtree | The subtree where the attribute uniqueness constraint is enforced. By default, it is enforced from the root directory. |
Garbage Collection Management Fields in Oracle Directory Manager

### Table A–9  Fields in the Garbage Collector Window

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Garbage Collector Name</td>
<td>You cannot modify this field.</td>
</tr>
<tr>
<td>Purge Base</td>
<td>The base DN of the naming context to which the garbage collection task is to be applied. You cannot modify this field.</td>
</tr>
<tr>
<td>Purge Debug</td>
<td>Indicator of whether to enable or disable debug logging for this garbage collector</td>
</tr>
<tr>
<td>Purge Enable Status</td>
<td>Enable or disable this garbage collector. The default is Enable.</td>
</tr>
<tr>
<td>Purge File Location</td>
<td>Absolute path name of the directory in which the log file is located</td>
</tr>
<tr>
<td>Purge File Name</td>
<td>Name of the log file</td>
</tr>
<tr>
<td>Purge Interval</td>
<td>The interval, in hours, after which the Garbage Collection job is executed again. For example, if you set this value to 12, then garbage collection occurs every 12 hours. This attribute is optional. The default value is 24.</td>
</tr>
<tr>
<td>Purge Now</td>
<td>Entering any value in this field means that, when you choose Apply, the garbage collection begins immediately. At that point, the value in this field automatically reverts to null.</td>
</tr>
<tr>
<td>Purge Start</td>
<td>Time, in seconds, when the Garbage collector runs for the first time. The format is YYYYMMDDHH24MISS. This attribute is optional. The default value is 0, which means that the garbage collector is enabled immediately.</td>
</tr>
<tr>
<td>Purge Target Age</td>
<td>Age, in hours, of the target objects. Objects older than the age specified in this attribute are purged at midnight. This attribute is optional. The default value is 12.</td>
</tr>
<tr>
<td>Purge Transaction Size</td>
<td>Number of objects to be purge in one committed transaction. This attribute is optional. The default value is 1000.</td>
</tr>
</tbody>
</table>

Oracle Internet Directory Statistics Collection Management Fields in Oracle Directory Manager

The Oracle Internet Directory statistics collector supports three of the modifiable fields described in Table A–9. They are Purge Interval, Purge Now, and Purge Start.

Password Policy Fields in Oracle Directory Manager

### Table A–10  Fields in the Password Policies General Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Must Supply Old Password When Modifying Password</td>
<td>Select whether the user must supply old password with new one when modifying password. By default, the old password is not required.</td>
</tr>
<tr>
<td>User Password Reversible Encryption</td>
<td>Select to store the password in a reversible encrypted format. By default, this option is disabled.</td>
</tr>
<tr>
<td>Reset Password upon Next Login</td>
<td>Select to have the password reset the next time the user logs in. By default, this option is disabled.</td>
</tr>
<tr>
<td>Old Password can be New Password</td>
<td>Allows the old password to be used as a new password. By default, this option is disabled.</td>
</tr>
</tbody>
</table>
Password Policy Fields in Oracle Directory Manager

Table A–10  (Cont.) Fields in the Password Policies General Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow hash consumption</td>
<td>Select to enables or disable logins using the hashed password value. By default, this option is enabled.</td>
</tr>
<tr>
<td>Grace Login Constraint</td>
<td>Select one of the following constraints for grace logins after a password has expired:</td>
</tr>
<tr>
<td></td>
<td>- None (default)</td>
</tr>
<tr>
<td></td>
<td>- Number of Grace Logins after Password Expiration</td>
</tr>
<tr>
<td></td>
<td>- Period for Grace Logins after Password Expiration</td>
</tr>
<tr>
<td>Number of Grace Logins after Password Expiration</td>
<td>Enter the maximum number of grace logins allowed after a password expires. The default value is 0. (By default, no grace logins are allowed.)</td>
</tr>
<tr>
<td>Period for Grace Logins after Password Expiration</td>
<td>Enter the number of seconds allowed for the grace logins after a password has expired. (By default, no grace logins are allowed.)</td>
</tr>
<tr>
<td>Password Expiry Time</td>
<td>Enter the number of seconds that a given password is valid. If this attribute is not present, or if the value is 0, then the password does not expire. By default, user passwords never expire.</td>
</tr>
<tr>
<td>Minimum Age for Password to Self-Modify</td>
<td>Enter the number of seconds before password expiration that the directory server sends the user a warning. If password expiration is enabled, then, by default, the directory server sends the user a warning three days before the password expires. The directory server sends the warning at each logon. If the user does not modify the password before it expires, then the directory server enforces the modification. This means that the user is locked out until the password is changed by the administrator. For this feature to work, the client application must support it.</td>
</tr>
<tr>
<td>Password Expiry Warning</td>
<td>Enter the name for this password policy.</td>
</tr>
</tbody>
</table>

Table A–11  Fields in the Password Policies Account Lockout Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Global Lockout Duration</td>
<td>Enter the number of seconds a user is locked out of the global directory if both of the following are true:</td>
</tr>
<tr>
<td></td>
<td>- Global lockout is enabled</td>
</tr>
<tr>
<td></td>
<td>- The user has been unable to bind successfully to the directory for at least the number of times specified by pwdMaxFailure</td>
</tr>
<tr>
<td></td>
<td>You can set user lockout for a specific duration, or until the administrator resets the user's password. The default value is 24 hours. A user account stays locked even after the lockout duration has passed unless the user binds with the correct password.</td>
</tr>
<tr>
<td>Password Maximum Failure</td>
<td>Enter the number of consecutive failed bind attempts after which a user account is locked.</td>
</tr>
<tr>
<td>Password Failure Count Interval</td>
<td>Enter the number of seconds after which the password failure times are purged from the user entry.</td>
</tr>
</tbody>
</table>
Table A–12 Fields in the Password Policies IP Lockout Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP Lockout Duration</td>
<td>Specify the number of seconds you want to enforce account lockout for a specific IP address.</td>
</tr>
<tr>
<td>IP Lockout Maximum Failure</td>
<td>Specify the maximum number of failed logins from a specific IP address after which the account is locked.</td>
</tr>
</tbody>
</table>

Table A–13 Fields in the Password Policies Password Syntax Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Numeric Characters</td>
<td>Specify the number of numeric characters required in a password.</td>
</tr>
<tr>
<td>Number of Passwords in History</td>
<td>Specify the maximum number of used passwords allowed.</td>
</tr>
<tr>
<td>Illegal Password Values</td>
<td>Enter values that you want disallowed in passwords.</td>
</tr>
<tr>
<td>Minimum Number of Characters of Password</td>
<td>Specify the minimum number of characters required in a password.</td>
</tr>
<tr>
<td>Minimum Number of Alphabetic Characters</td>
<td>Specify the minimum number of alphabetic characters required in a password.</td>
</tr>
<tr>
<td>Minimum Number of Special Characters</td>
<td>Specify the minimum number of characters that are not alphanumeric (special characters), such as %, #, $, @, required in a password.</td>
</tr>
<tr>
<td>Minimum Number of Uppercase Characters</td>
<td>Specify the minimum number of uppercase characters required in a password.</td>
</tr>
<tr>
<td>Minimum Number of Lowercase Characters</td>
<td>Specify the minimum number of lowercase characters required in a password.</td>
</tr>
<tr>
<td>Maximum Number of Repeated Characters</td>
<td>Specify the maximum number of repeated characters required in a password.</td>
</tr>
</tbody>
</table>

Password Verifier Fields in Oracle Directory Manager

Table A–14 Fields in the Password Verifier Profile Dialog Box

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Path to Password Verifier Entry</td>
<td>The full DN of this password verifier entry. Use this to locate a particular password verifier entry. You cannot modify this field.</td>
</tr>
<tr>
<td>Password Verifier Entry</td>
<td>RDN of this password verifier. You cannot modify this field.</td>
</tr>
<tr>
<td>Owner</td>
<td>The DN of the administrator of the verifier entry. You can modify this field.</td>
</tr>
<tr>
<td>Application ID</td>
<td>The unique identifier of the Oracle application. It is generated during application installation. You cannot modify this field.</td>
</tr>
</tbody>
</table>
Plug-in Management Fields in Oracle Directory Manager

Oracle Password Parameters
Parameters containing information for generating this password verifier. Use this field to specify the hashing algorithm for this password verifier. The syntax is:

crypto:hashing_algorithm
For example, if you are using the ORCLLM hashing algorithm, then you would enter:
crypto:ORCLLM
If you are using SASL/MD5, for example, you can enter the following:
crypto:SASL/MD5 $ realm:dc=com

Plug-in Management Fields in Oracle Directory Manager

See Also: "Registering and Managing Plug-ins" on page 32-4

Table A–15 Fields in the New Plug-in Dialog Box, Mandatory Properties tab page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plug-in Enable</td>
<td>Either disabled (default) or enabled.</td>
</tr>
<tr>
<td>Plug-in Is Replacement</td>
<td>Either disabled (default) or enabled. For when_replace timing, select Enable and then set Plug-in Timing to WHEN.</td>
</tr>
<tr>
<td>Plug-in Package Name</td>
<td>Name of the package for this plug-in.</td>
</tr>
</tbody>
</table>
| Plug-in Type         | Operational. Operation plug-ins augment existing LDAP operations. The work they perform depends on whether they execute before, after, or in addition to normal directory server operations. See Also: Chapter 32, "Oracle Internet Directory Server Plug-in Framework"
| Plug-in Kind         | PL/SQL or Java.                                                            |
| Plug-in LDAP Operation| One of the following values:
  - ldapadd
  - ldapbind
  - ldapcompare
  - ldapdelete
  - ldapmodify
  - ldapsearch
  - ldapmoddn (Java only) |
| Plug-in Timing       | One of the following values:
  - pre—for plug-ins that the directory server calls before performing an LDAP operation
  - when—for plug-ins that the directory server calls in addition to standard processing of an LDAP operation
  - post—for plug-ins that the directory server calls after performing an LDAP operation
### Table A–16  Fields in the New Plug-in Dialog Box, Optional Properties Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plug-in Class Reload Enabled</td>
<td>Either enabled (default) or disabled.</td>
</tr>
<tr>
<td>Plug-in Version</td>
<td>Supported plug-in version number.</td>
</tr>
<tr>
<td>Plug-in Subscriber DN List</td>
<td>A semicolon separated DN list that controls if the plug-in takes effect. For example:</td>
</tr>
<tr>
<td></td>
<td><code>orclPluginSubscriberDNList=dc=COM,c=us;dc=us,dc=oracle,dc=com;dc=org,dc=us;o=IMC,c=US</code></td>
</tr>
<tr>
<td></td>
<td>The target DN of an LDAP operation is included in the list, then the plug-in is invoked.</td>
</tr>
<tr>
<td>Plug-in Attribute List</td>
<td>A list of semicolon-separated attribute names that controls whether the plug-in takes effect. If the target attribute is included in the list, then the plug-in is invoked. Only for ldapcompare and ldapmodify plug-ins.</td>
</tr>
<tr>
<td>Plug-in Result Code</td>
<td>An integer value to specify the LDAP result code. If this value is specified, then plug-in will be invoked only if the LDAP operation is in that result code scenario. Only for the POST plug-in type.</td>
</tr>
<tr>
<td>Plug-in Entry Properties</td>
<td>An LDAP search filter type. For example, if you specify the following, the plug-in is not invoked if the target entry has <code>objectclass</code> equal to <code>inetorgperson</code> and <code>sn</code> equal to Cezanne: <code>orclPluginEntryProperties:(&amp;(objectclass=inetorgperson)(sn=Cezanne))</code></td>
</tr>
<tr>
<td></td>
<td>When you click the Browse button, you can filter the entry properties. Using this example, in the Entry Filter dialog box, you would enter <code>inetorgperson</code> and Cezanne as the entry properties criteria.</td>
</tr>
<tr>
<td>Plug-in Request Group</td>
<td>A semicolon-separated group list that controls if the plug-in takes effect. You can use this group to specify who can actually invoke the plug-in. For example, if you specify <code>orclpluginrequestgropu:cn=security,cn=groups,dc=oracle,dc=com</code> when you register the plug-in the plug-in will not be invoked unless the LDAP request comes from the person who belongs to the group <code>cn=security,cn=groups,dc=oracle,dc=com</code>.</td>
</tr>
<tr>
<td>Plug-in Binary Flex Field</td>
<td>The custom binary information passed to the Java plug-in. This is a file location, for example, <code>/home/user1/pic/mypicture.jpg</code>.</td>
</tr>
<tr>
<td>Flex Fields</td>
<td>Allows you to create your own custom flex field values.</td>
</tr>
</tbody>
</table>

### Table A–17  Fields in the Edit Plug-in Dialog Box, Mandatory Properties Tab Page:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plug-in Enable</td>
<td>Either disabled (default) or enabled.</td>
</tr>
</tbody>
</table>
Plug-in Management Fields in Oracle Directory Manager

Table A–17  Fields in the Edit Plug-in Dialog Box, Mandatory Properties tab page:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plug-in Is Replacement</td>
<td>Either disabled (default) or enabled.</td>
</tr>
<tr>
<td></td>
<td>For when_replace timing, select Enable and then set Plug-in Timing to WHEN.</td>
</tr>
<tr>
<td>Plug-in Package Name</td>
<td>Name of the package for this plug-in.</td>
</tr>
<tr>
<td>Plug-in Type</td>
<td>Operational. Operation plug-ins augment existing LDAP operations. The work they perform depends on whether they execute before, after, or in addition to normal directory server operations. See Also: Chapter 32, &quot;Oracle Internet Directory Server Plug-in Framework&quot;</td>
</tr>
<tr>
<td>Plug-in Kind</td>
<td>PL/SQL or Java.</td>
</tr>
<tr>
<td>Plug-in LDAP Operation</td>
<td>One of the following values:</td>
</tr>
<tr>
<td></td>
<td>■ ldapadd</td>
</tr>
<tr>
<td></td>
<td>■ ldapbind</td>
</tr>
<tr>
<td></td>
<td>■ ldapcompare</td>
</tr>
<tr>
<td></td>
<td>■ ldapdelete</td>
</tr>
<tr>
<td></td>
<td>■ ldapmodify</td>
</tr>
<tr>
<td></td>
<td>■ ldapsearch</td>
</tr>
<tr>
<td></td>
<td>■ ldapmoddn (Java only)</td>
</tr>
<tr>
<td>Plug-in Timing</td>
<td>One of the following values:</td>
</tr>
<tr>
<td></td>
<td>■ pre—for plug-ins that the directory server calls before performing an LDAP operation</td>
</tr>
<tr>
<td></td>
<td>■ when—for plug-ins that the directory server calls in addition to standard processing of an LDAP operation</td>
</tr>
<tr>
<td></td>
<td>■ post—for plug-ins that the directory server calls after performing an LDAP operation</td>
</tr>
</tbody>
</table>

Table A–18  Fields in the Edit Plug-in Dialog Box, Optional Properties tab page:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plug-in Class Reload</td>
<td>Either enabled (default) or disabled.</td>
</tr>
<tr>
<td>Enabled</td>
<td>Supported plug-in version number.</td>
</tr>
</tbody>
</table>
| Plug-in Version        | A semicolon separated DN list that controls if the plug-in takes effect. For example:  
                          | orclPluginSubscriberDNList=dc=COM,c=us;dc=us,dc=oracle,dc=com;dc=org,dc=us;o=IMC,c=US |
                          | The target DN of an LDAP operation is included in the list, then the plug-in is invoked. |
| Plug-in Subscriber DN List | A list of semicolon-separated attribute names that controls whether the plug-in takes effect. If the target attribute is included in the list, then the plug-in is invoked. Only for ldapcompare and ldapmodify plug-ins. |
Replication Fields in Oracle Directory Manager

Table A–18 (Cont.) Fields in the Edit Plug-in Dialog Box, Optional Properties tab page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Plug-in Result Code</td>
<td>An integer value to specify the LDAP result code. If this value is specified, then plug-in will be invoked only if the LDAP operation is in that result code scenario. Only for the POST plug-in type.</td>
</tr>
<tr>
<td>Plug-in Entry Properties</td>
<td>An LDAP search filter type. For example, if you specify the following, the plug-in is not invoked if the target entry has objectclass equal to inetorgperson and sn equal to Cezanne: orclPluginEntryProperties:(&amp;(objectclass=inetorgperson)(sn=Cezanne)) When you click the Browse button, you can filter the entry properties. Using this example, in the Entry Filter dialog box, you would enter inetorgperson and Cezanne as the entry properties criteria.</td>
</tr>
<tr>
<td>Plug-in Request Group</td>
<td>A semicolon-separated group list that controls if the plug-in takes effect. You can use this group to specify who can actually invoke the plug-in. For example, if you specify orclpluginrequestgroup:cn=security,cn=groups,dc=oracle,dc=com when you register the plug-in the plug-in will not be invoked unless the LDAP request comes from the person who belongs to the group cn=security,cn=groups,dc=oracle,dc=com.</td>
</tr>
<tr>
<td>Plug-in Binary Flex Field</td>
<td>The custom binary information passed to the Java plug-in. This is a file location, for example, /home/user1/pic/mypicture.jpg.</td>
</tr>
<tr>
<td>Flex Fields</td>
<td>Allows you to create your own custom flex field values.</td>
</tr>
</tbody>
</table>

Table A–19 Fields in the Edit Plug-in Dialog Box, Advanced tab page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attribute</td>
<td>Enter an attribute not shown on the Optional or Mandatory properties tabs.</td>
</tr>
<tr>
<td>Attribute Options</td>
<td>Enter options for the attribute.</td>
</tr>
<tr>
<td>Attribute Value</td>
<td>Enter a value for the attribute.</td>
</tr>
</tbody>
</table>

Replication Fields in Oracle Directory Manager

Table A–20 Fields in the Replication Server Configuration Set: General Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Orclnormdn</td>
<td>The normalized DN of the configuration set entry.</td>
</tr>
<tr>
<td>Change Retry Count</td>
<td>Enter the number of attempts that the conflict resolution process tries to apply each update before giving up and logging the incident. The default is 10. You can modify this field.</td>
</tr>
</tbody>
</table>
Dump flags
Determines whether to generate a stack file (default value 0) or OS level core file (value 1) if the server crashes.

Change Log Life
Enter the number of hours for the life of the change log objects. You can modify this field.

Number of Threads to Transport per Supplier
Enter the number of worker threads the directory replication server provides for each supplier for transport. The default is 1. You can modify this field.

Number of Threads to Apply per Supplier
Enter the number of worker threads the directory replication server provides for each supplier for change log processing. The default is 5. You can modify this field.

Set
The configuration identifier. You cannot modify this field.

See Also: "Modifying Configuration Parameters of the Directory Replication Server by Using Oracle Directory Manager" on page 31-2

Table A–21 Fields in the ASR Agreement Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Consumer Replica DN</td>
<td>This attribute specifies the DN of the replica to identify a consumer in the replication agreement. You cannot modify this attribute.</td>
</tr>
<tr>
<td>Excluded Naming Contexts</td>
<td>This attribute lists all the naming contexts that are excluded from replication on a multimaster replica. It is not applicable to an LDAP-based replica.</td>
</tr>
<tr>
<td>HIQ Schedule</td>
<td>The interval, in minutes, at which the directory replication server repeats the change application process. You can modify this field.</td>
</tr>
<tr>
<td>Replication Group Nodes</td>
<td>For Advanced Replication-based groups, enter the orclreplicaids values of all the nodes in this replication group. This list must be identical on all nodes in the group. This attribute is not applicable to LDAP-based replication agreements.</td>
</tr>
<tr>
<td>Keep LDAP Connection Alive</td>
<td>This attribute determines whether connections from the directory replication server to the directory server are kept active or established every time the changelog processing is done based on various schedules. You can modify this field.</td>
</tr>
<tr>
<td>Replica Agreement ID</td>
<td>Naming attribute for the replication agreement entry. Values:</td>
</tr>
<tr>
<td>Replica Agreement Protocol</td>
<td>This attribute defines the replication protocol for change propagation to the replica. Values:</td>
</tr>
<tr>
<td></td>
<td>• ODS_ASR_1.0 (Advanced Replication-based replication)</td>
</tr>
<tr>
<td></td>
<td>• ODS_LDAP_1.0 (LDAP-based replication)</td>
</tr>
<tr>
<td>Update Schedule</td>
<td>This attribute specifies the replication update interval for new changes and those being retried. The value is in minutes. You can modify this field.</td>
</tr>
</tbody>
</table>
### Table A–22  Fields in the Replica Node: General Tab Page

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Replica URI</td>
<td>Contains information in ldapURI format that can be used to open a connection to this replica.</td>
</tr>
<tr>
<td>Replica Secondary URI</td>
<td>Contains the set of ldapURI format addresses that can be used if the orclReplicaURI values cannot be used.</td>
</tr>
</tbody>
</table>
| Current Replica State   | A read-only attribute that defines the current state of the replica. Possible values are:  
  - Online  
  - Offline  
  - Bootstrap  
  - Bootstrap in Progress  
  - Bootstrap Error  
  - Bootstrap Done  
  - DB Copy Add Mode |
| Replica State           | Specifies the desired state of the replica. Possible values are:  
  - Online  
  - Offline  
  - Bootstrap |
| Replica Type            | Defines the type of replica such as read-only or read/write. Possible values:  
  - 0 (Read/Write)  
  - 1 (Read-Only) |

### Table A–23  Columns in the Replica Agreements Tab Page

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
</table>
| Consumer Replica DN     | This attribute specifies the DN of the replica to identify a consumer in the replication agreement.  
  You cannot modify this field. |
| HIQ Schedule            | The interval, in minutes, at which the directory replication server repeats the change application process. You can modify this field. |
| Keep LDAP Connection Alive | This attribute determines whether connections from the directory replication server to the directory server are kept active or established every time the changelog processing is done based on various schedules. You can modify this field. |
| Last Applied Change Number | This attribute indicates the status of the consumer replica with respect to the supplier in an LDAP-based replication agreement.  
  This attribute is not applicable for Advanced Replication-based agreements. |
| Replica Agreement ID    | Naming attribute for the replication agreement entry. |
| Replication Protocol    | This attribute defines the replication protocol for change propagation to the replica.  
  Values:  
  - ODS_ASR_1.0 (Advanced Replication-based replication)  
  - ODS_LDAP_1.0 (LDAP-based replication) |
Replication Fields in Oracle Directory Manager

**Table A–23  (Cont.) Columns in the Replica Agreements Tab Page**

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Update Schedule</td>
<td>This attribute defines the replication update interval for new changes and those being retried. The value is in minutes. You can modify this field.</td>
</tr>
</tbody>
</table>

**Table A–24  Fields in the Replica Agreement: Replica Naming Context Tab Page**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Excluded Attributes</td>
<td>For partial replication only. Within the included naming context, an attribute to be excluded from replication. This is a multivalued attribute.</td>
</tr>
<tr>
<td>Excluded Naming Contexts</td>
<td>The root of a subtree to be excluded from replication. This is a multivalued attribute. You can modify this field. For LDAP-based replication, from within the naming context specified in the <code>orclincludednamingcontexts</code> attribute, you can specify one or more subtrees in the LDAP naming context object so that they are excluded from partial replication. For replication agreements based on Advanced Replication, you can specify one or more subtrees to be excluded from replication.</td>
</tr>
<tr>
<td>Included Naming Contexts</td>
<td>The naming context included in a partial replica. This is a single valued attribute. For each naming context object, you can specify only one unique subtree. In partial replication, except for subtrees listed in the <code>orclexcludednamingcontexts</code> attribute, all subtrees in the specified included naming context are replicated. <strong>Note:</strong> Only LDAP-based replication agreements respect this attribute to define one or more partial replicas. If this attribute contains any values in an Advanced Replication-based replication agreement, then it is ignored. You can modify this attribute.</td>
</tr>
<tr>
<td>Synchronize Filter</td>
<td>This check box appears only if this is a two-way replica agreement. When the check box is selected (the default) the configuration is the same in both directions. When this check box is not selected, an additional Naming Context tab page appears. The new page allows you to separately configure the naming context in the other direction.</td>
</tr>
</tbody>
</table>

**See Also:** "Determining What Is to Be Replicated in LDAP-Based Partial Replication" on page 30-33

**Table A–25  Fields in the NewReplica Agreement: Naming Context Tab Page**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Excluded Attributes</td>
<td>For partial replication only. Within the included naming context, an attribute to be excluded from replication. This is a multivalued attribute.</td>
</tr>
</tbody>
</table>
Replication Fields in Oracle Directory Manager

**Table A–25 (Cont.) Fields in the NewReplica Agreement: Naming Context Tab Page**

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Excluded Naming Contexts</td>
<td>The root of a subtree to be excluded from replication. This is a multivalued attribute. You can modify this field. For LDAP-based replication, from within the naming context specified in the <code>orclincludednamingcontexts</code> attribute, you can specify one or more subtrees in the LDAP naming context object so that they are excluded from partial replication. For replication agreements based on Advanced Replication, you can specify one or more subtrees to be excluded from replication.</td>
</tr>
</tbody>
</table>

| Included Naming Contexts | The naming context included in a partial replica. This is a single valued attribute. For each naming context object, you can specify only one unique subtree. In partial replication, except for subtrees listed in the `orclexcluednamingcontexts` attribute, all subtrees in the specified included naming context are replicated. **Note:** Only LDAP-based replication agreements respect this attribute to define one or more partial replicas. If this attribute contains any values in an Advanced Replication-based replication agreement, then it is ignored. |

**Table A–26 Columns in the Replica Agreement: Window**

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agreement Type</td>
<td>A read-only field. Possible values are: ■ One-way ■ Two-way</td>
</tr>
<tr>
<td>Consumer Replica DN</td>
<td>This attribute specifies the DN of the replica to identify a consumer in the replication agreement. You cannot modify this field.</td>
</tr>
<tr>
<td>Excluded Naming Contexts</td>
<td>This attribute lists all the naming contexts that are excluded from replication on a multimaster replica. It is not applicable to an LDAP-based replica.</td>
</tr>
<tr>
<td>HIQ Schedule</td>
<td>The interval, in minutes, at which the directory replication server repeats the change application process. You can modify this field.</td>
</tr>
<tr>
<td>Included Naming Contexts</td>
<td>This attribute is required to list all the naming contexts that are included on the partial replica. <strong>Note:</strong> Only LDAP-based replication agreements respect these attributes to define partial replicas. If this attribute contains any values in an Oracle Database Advanced Replication-based agreement, then it will be ignored. You can modify this field.</td>
</tr>
<tr>
<td>Keep LDAP Connection Alive</td>
<td>This attribute determines whether connections from the directory replication server to the directory server are kept active or established every time the changelog processing is done based on various schedules. You can modify this field.</td>
</tr>
<tr>
<td>Replica Agreement ID</td>
<td>Naming attribute for the replication agreement entry.</td>
</tr>
<tr>
<td>Replica Agreement Protocol</td>
<td>This attribute defines the replication protocol for change propagation to the replica. Values: ■ ODS_ASR_1.0 (Advanced Replication-based replication) ■ ODS_LDAP_1.0 (LDAP-based replication)</td>
</tr>
</tbody>
</table>
This section contains these topics:

- **Object Classes Fields in Oracle Directory Manager**
- **Attributes Fields in Oracle Directory Manager**
- **Matching Rules Fields in Oracle Directory Manager**
- **Content Rules Management Fields in Oracle Directory Manager**

### Object Classes Fields in Oracle Directory Manager

This attribute specifies the replication update interval for new changes and those being retried. The value is in minutes. You can modify this field.

For a one-way agreement type, this table shows one row of data. For a two-way agreement type, it shows two rows of data. The table indicates the supplier and consumer nodes, the last applied change, and the last transported change.

### Table A–26 (Cont.) Columns in the Replica Agreement: Window

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Update Schedule</td>
<td>This attribute specifies the replication update interval for new changes and those being retried. The value is in minutes. You can modify this field.</td>
</tr>
<tr>
<td>Replication Status</td>
<td>For a one-way agreement type, this table shows one row of data. For a two-way agreement type, it shows two rows of data. The table indicates the supplier and consumer nodes, the last applied change, and the last transported change.</td>
</tr>
</tbody>
</table>

### Table A–27 Fields in the Change Log Window

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Change Log Number</td>
<td>The unique identifier of this change</td>
</tr>
<tr>
<td>Change Log Operation</td>
<td>The type of operation that this change effected—for example, add, modify, delete, compare</td>
</tr>
<tr>
<td>Change Log Target DN</td>
<td>The DN of the entry upon which this change was effected</td>
</tr>
<tr>
<td>Change Log Target DN Changes</td>
<td>The changes made to the entry</td>
</tr>
<tr>
<td>Change Retry Count</td>
<td>The number of attempts to apply this change to another node in a replicated environment</td>
</tr>
<tr>
<td>Modifier's Name</td>
<td>The name of the user who effected the change</td>
</tr>
<tr>
<td>Operation Time</td>
<td>The time at which the change took place</td>
</tr>
<tr>
<td>Orcl GUID</td>
<td>The global unique identifier of the entry on which the change is made</td>
</tr>
<tr>
<td>Orcl Parent GUID</td>
<td>The global unique identifier of the parent of the entry on which the change is made</td>
</tr>
<tr>
<td>Server Name</td>
<td>The name of the server from which the change was issued</td>
</tr>
</tbody>
</table>

See Also: Chapter 11, "Directory Schema Administration"

### Object Classes Fields in Oracle Directory Manager

This section contains these topics:

- **Object Classes Fields in Oracle Directory Manager**
- **Attributes Fields in Oracle Directory Manager**
- **Matching Rules Fields in Oracle Directory Manager**
- **Content Rules Management Fields in Oracle Directory Manager**

### Table A–28 Object Class Properties Listed in Searches in Oracle Directory Manager

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the object class for which you are searching. For example, the phrase Name Exact Match subAc1 gives you the subAc1 object class.</td>
</tr>
</tbody>
</table>
Object ID

Object identifier for the object class for which you are searching. For example, the phrase Object ID Begins With 2.5.2 gives you a list of object classes whose object identifiers begin with 2.5.2.

The object identifier is a standardized numerical sequence based on IETF standards. It must be unique, and should comply with the system established within your organization. Normally it is derived from the identifier assigned by registration agencies, such as ANSI or ISO.

Description

Word in the description field. For example, the phrase Description Contains Shoe gives you a list of object classes with the word shoe in the description column. This field is optional, for your information only.

Type

Type of object class for which you are searching, whether abstract, structural, or auxiliary.

Super Class

Class from which the object class for which you are searching is derived. Clicking Add displays the Super Class Selector dialog box from which you can select the superclass(es) you want to add.

Mandatory Attributes

Mandatory attributes of the object class for which you are searching. For example, the phrase Mandatory Attributes Contains cn gives you a list of all object classes in which the cn attribute is mandatory.

Optional Attributes

Optional attributes of the object class for which you are searching.

Table A–28

Object Class Properties Listed in Searches in Oracle Directory

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Object ID</td>
<td>Object identifier for the object class for which you are searching. For example, the phrase Object ID Begins With 2.5.2 gives you a list of object classes whose object identifiers begin with 2.5.2. The object identifier is a standardized numerical sequence based on IETF standards. It must be unique, and should comply with the system established within your organization. Normally it is derived from the identifier assigned by registration agencies, such as ANSI or ISO.</td>
</tr>
<tr>
<td>Description</td>
<td>Word in the description field. For example, the phrase Description Contains Shoe gives you a list of object classes with the word shoe in the description column. This field is optional, for your information only.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of object class for which you are searching, whether abstract, structural, or auxiliary.</td>
</tr>
<tr>
<td>Super Class</td>
<td>Class from which the object class for which you are searching is derived. Clicking Add displays the Super Class Selector dialog box from which you can select the superclass(es) you want to add.</td>
</tr>
<tr>
<td>Mandatory Attributes</td>
<td>Mandatory attributes of the object class for which you are searching. For example, the phrase Mandatory Attributes Contains cn gives you a list of all object classes in which the cn attribute is mandatory.</td>
</tr>
<tr>
<td>Optional Attributes</td>
<td>Optional attributes of the object class for which you are searching.</td>
</tr>
</tbody>
</table>

Table A–29

Search Filters for Object Classes

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Begins With</td>
<td>Searches by using only the first few characters of the property of the object class for which you are searching. For example, the phrase Type Begins With aux gives you a list of all of the auxiliary object classes.</td>
</tr>
<tr>
<td>Ends With</td>
<td>Searches by using only the last few characters of the property of the object class for which you are searching. For example, the phrase Type Ends With ral gives you a list of all of the structural object classes.</td>
</tr>
<tr>
<td>Contains</td>
<td>Searches for object classes in which the property you selected includes, but is not necessarily limited to, the value you enter. For example, the phrase Optional Attributes Contains cn gives you a list of all object classes in which cn is an optional attribute.</td>
</tr>
<tr>
<td>Exact Match</td>
<td>Searches for an object class in which the property you selected is exactly the same as the value you enter. For example, the phrase Super Class Exact Match person gives you a list of all object classes that have person as their superclass.</td>
</tr>
<tr>
<td>Greater Or Equal</td>
<td>Searches for an object class in which the property you selected is numerically or alphabetically greater than or equal to the value you enter. For example, the phrase Name Greater or Equal orcl gives you a list of object classes from those beginning with the letters orcl to those beginning with letters at the end of the alphabet.</td>
</tr>
<tr>
<td>Less Or Equal</td>
<td>Searches for an object class in which the property you selected is numerically or alphabetically less than or equal to the value you enter. For example, the phrase Name Less Or Equal orcl gives you a list of object classes from those beginning with the letters orcl to those at the beginning of the alphabet.</td>
</tr>
<tr>
<td>Not Null</td>
<td>Searches for all object classes in which the property you selected is present. For example, the phrase Mandatory Attributes Not Null gives you a list of all object classes which contain mandatory attributes.</td>
</tr>
</tbody>
</table>
### Table A–30 Buttons Used in Searches for Object Classes in Oracle Directory Manager

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>Creates a new search criteria bar in the Criteria field. This button is enabled only when the Criteria bar has been deleted.</td>
</tr>
<tr>
<td>And</td>
<td>Creates another search criteria bar in the Criteria field. Matches all object classes having one specified criterion with those that also have another specified criterion.</td>
</tr>
<tr>
<td>Or</td>
<td>Creates another search criteria bar in the Criteria field. Matches all object classes with either one specified attribute or another.</td>
</tr>
<tr>
<td>Not</td>
<td>Negates the criterion in the selected search criteria bar and retrieves all object classes that do not have the specified criterion.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes a selected search criteria bar</td>
</tr>
</tbody>
</table>

### Table A–31 Fields in the New Object Class Dialog Box

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the object class.</td>
</tr>
<tr>
<td>Object ID</td>
<td>Object identifier. This is a standardized numerical sequence based on IETF standards. It must be unique, and should comply with the system established within your organization. Normally it is derived from the identifier assigned by registration agencies, such as ANSI or ISO.</td>
</tr>
<tr>
<td>Description</td>
<td>Use this optional field for your information only.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of object class: Abstract, Structural, Auxiliary, None.</td>
</tr>
<tr>
<td>Super Class</td>
<td>Class(es) from which to derive this object class. This object class will inherit all the attributes of the superclass(es) you select. Every structural object class must have top as one of its superclasses. Clicking Add displays the Super Class Selector dialog box from which you can select the superclass(es) you want to add.</td>
</tr>
<tr>
<td>Mandatory Attributes</td>
<td>Attributes for which values must be entered. Clicking Add displays the Mandatory Attributes Selector dialog box from which you can select the mandatory attributes you want to add.</td>
</tr>
<tr>
<td>Optional Attributes</td>
<td>Attributes for which values are not required. Clicking Add displays the Optional Attributes Selector dialog box from which you can select the optional attributes you want to add.</td>
</tr>
</tbody>
</table>

### Attributes Fields in Oracle Directory Manager

#### Table A–32 Columns in the Attributes Tab Page in Oracle Directory Manager

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The standardized attribute type names</td>
</tr>
<tr>
<td>Indexed</td>
<td>Check boxes indicating whether attributes are indexed</td>
</tr>
<tr>
<td>Object ID</td>
<td>Standardized object identifier for each attribute</td>
</tr>
<tr>
<td>Description</td>
<td>Words describing each attribute</td>
</tr>
<tr>
<td>Syntax</td>
<td>The standardized rules for data entry applicable to each attribute type</td>
</tr>
<tr>
<td>Size</td>
<td>Maximum size allowed for each object</td>
</tr>
</tbody>
</table>
**Table A–32** (Cont.) Columns in the Attributes Tab Page in Oracle Directory Manager

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
</table>
| Usage  | Standards specifying how the attribute can be used. There are four options:  
  - userApplications  
  - directoryOperation  
  - distributedOperation  
  - dSAOperation. |
| Ordering | Standards specifying how precedence is established for values |
| Equality | Standards specifying how equality is determined in compare and search operations |
| Substring | Regular expression matching string |
| Single Value | Attribute types containing a maximum of one value |
| Super | Super attribute for each attribute |

**Table A–33** Search Filters for Attributes

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Begins With</td>
<td>Searches by using only the first few characters of the property's value. For example, the phrase Syntax Begins With 1.3 gives you a list of all attributes in which the first few numbers of the syntax identifier are 1.3.</td>
</tr>
<tr>
<td>Ends With</td>
<td>Searches by using only the last few characters of the property's value. For example, the phrase Name Ends With License gives you a list of all attributes with that ending, such as carLicense.</td>
</tr>
<tr>
<td>Contains</td>
<td>Searches for attributes that include the property with the value you enter. For example, the phrase Ordering Contains time gives you a list of all attributes with the word time in the Ordering column.</td>
</tr>
<tr>
<td>Exact Match</td>
<td>Searches for a value that is exactly the same as that found in the attribute property you specified. For example, the phrase Equality Exact Match caseIgnoreMatch gives you a list of all attributes that have the caseIgnoreMatch matching rule.</td>
</tr>
<tr>
<td>Greater or Equal</td>
<td>Searches for an attribute that has a property that is numerically or alphabetically greater than or equal to the value you enter. For example, the phrase Name Greater or Equal orcl gives you a list of attributes from those beginning with orcl to those beginning with letters at the end of the alphabet.</td>
</tr>
<tr>
<td>Less or Equal</td>
<td>Searches for an attribute that has a property that is numerically or alphabetically less than or equal to the value you enter. For example, the phrase Name Less or Equal orcl gives you a list of attributes from those beginning with orcl to those beginning with letters at the start of the alphabet.</td>
</tr>
<tr>
<td>Not Null</td>
<td>Searches for all attributes in which the attribute property you selected is present. For example, the phrase Description Not Null gives you a list of all attributes which have text in the description field.</td>
</tr>
</tbody>
</table>

**Table A–34** Buttons in Searches for Attributes in Oracle Directory Manager

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>Creates a new search criteria bar in the Criteria field. This button is enabled only when the Criteria field is empty.</td>
</tr>
<tr>
<td>And</td>
<td>Creates another search criteria bar in the Criteria field. Matches all attributes with one specified property with those that also have another specified property.</td>
</tr>
</tbody>
</table>
Creates another search criteria bar in the Criteria field. Matches all attributes with either one specified property or another.

Negates the criteria in the selected search criteria bar and matches all attributes that do not have the property specified.

Deletes a selected search criteria bar

<table>
<thead>
<tr>
<th>Table A–35</th>
<th>Fields in the General Tab Page of the New Attribute Type Dialog</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>Name</td>
<td>Name for this attribute</td>
</tr>
<tr>
<td>Object ID</td>
<td>Object ID for this attribute. The Object ID is a standardized numerical sequence based on IETF standards. It must be unique. Normally this is derived from the identifier assigned by registration agencies, such as ANSI or ISO. For an explanation of the standard identifiers, see the current LDAP standards available through the IETF Web site at <a href="http://www.ietf.org">http://www.ietf.org</a>.</td>
</tr>
<tr>
<td>Description</td>
<td>Optional field for your information only</td>
</tr>
<tr>
<td>Syntax</td>
<td>Standardized rules for data entry applicable to this attribute type</td>
</tr>
<tr>
<td>Size</td>
<td>Maximum size allowed for this object</td>
</tr>
<tr>
<td>Single Value</td>
<td>Indicator that this attribute type contains a maximum of one value.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Table A–36</th>
<th>Fields in the Advanced Tab Page of the New Attribute Type Dialog</th>
</tr>
</thead>
<tbody>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>Indexed</td>
<td>Select this box to add the attribute to the index, thereby making it available for use in a search. Only those attributes that have an equality matching rule can be indexed.</td>
</tr>
<tr>
<td>Usage</td>
<td>Specify standards for how the attribute can be used. Options are:</td>
</tr>
<tr>
<td></td>
<td>■ userApplications</td>
</tr>
<tr>
<td></td>
<td>Attributes whose values must be entered by the user, for example,</td>
</tr>
<tr>
<td></td>
<td>telephoneNumber</td>
</tr>
<tr>
<td></td>
<td>■ directoryOperation</td>
</tr>
<tr>
<td></td>
<td>Attributes whose values are entered by the directory server, for example,</td>
</tr>
<tr>
<td></td>
<td>creatorName or timeStamp</td>
</tr>
<tr>
<td></td>
<td>■ distributedOperation</td>
</tr>
<tr>
<td></td>
<td>■ dSAOperation</td>
</tr>
<tr>
<td></td>
<td>Attributes used for the internal operation of the server, for example,</td>
</tr>
<tr>
<td></td>
<td>orclUpdateSchedule</td>
</tr>
<tr>
<td>Ordering</td>
<td>Specify standards for how precedence is established for values.</td>
</tr>
<tr>
<td>Equality</td>
<td>Specify standards for how equality is determined in compare and search operations.</td>
</tr>
<tr>
<td>Substring</td>
<td>Specify the matching rule.</td>
</tr>
</tbody>
</table>
Super

Add the super attribute for this attribute. To do this:

1. Choose the Add button next to this field. The Super Attribute Selector appears.
2. Select the super attribute and choose Select.
3. Repeat as needed.

To delete a super attribute from the Super field, select it, then choose Delete.

Matching Rules Fields in Oracle Directory Manager

<table>
<thead>
<tr>
<th>Column Head</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the attribute matching rule</td>
</tr>
<tr>
<td>Object ID</td>
<td>Unique identifier of this matching rule</td>
</tr>
<tr>
<td>Description</td>
<td>Words describing the matching rule (optional)</td>
</tr>
<tr>
<td>Syntax</td>
<td>Syntax used with this matching rule</td>
</tr>
</tbody>
</table>

Content Rules Management Fields in Oracle Directory Manager

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structural Object Class</td>
<td>The name of the structural object class to which you want to assign this content rule</td>
</tr>
<tr>
<td>Object ID</td>
<td>The unique identifier of the content rule you are creating</td>
</tr>
<tr>
<td>Label</td>
<td>A descriptive friendly name of this content rule</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td>The auxiliary object classes whose attributes you want to associate with the specified structural object class. To specify an auxiliary class:</td>
</tr>
<tr>
<td></td>
<td>1. Choose Add. The Auxiliary Class Selector dialog box appears.</td>
</tr>
<tr>
<td></td>
<td>2. Select the auxiliary class you want to add.</td>
</tr>
<tr>
<td></td>
<td>3. Choose Select. This returns you to the New Content Rule dialog box. The auxiliary class you just specified appears in the Auxiliary Classes field.</td>
</tr>
<tr>
<td>Mandatory Attributes</td>
<td>The mandatory attributes you want to associate with the specified structural object class. To specify a mandatory attribute:</td>
</tr>
<tr>
<td></td>
<td>2. Select the mandatory attribute you want to add. If you want this attribute indexed, then select the corresponding check box in the Indexed column.</td>
</tr>
<tr>
<td></td>
<td>3. Choose Select. This returns you to the New Content Rule dialog box. The mandatory attribute you just specified appears in the Mandatory Attributes field.</td>
</tr>
</tbody>
</table>
### Table A–38 (Cont.) Fields in the New Content Rule Dialog Box

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optional Attributes</td>
<td>The optional attributes you want to associate with the specified structural object class. To specify an optional attribute:</td>
</tr>
<tr>
<td></td>
<td>1. Choose Add. The Optional Attribute Selector dialog box appears.</td>
</tr>
<tr>
<td></td>
<td>2. Select the optional attribute you want to add. If you want this attribute indexed, then select the corresponding check box in the Indexed column.</td>
</tr>
<tr>
<td></td>
<td>3. Choose Select. This returns you to the New Content Rule dialog box. The optional attribute you just specified appears in the Optional Attributes field.</td>
</tr>
</tbody>
</table>

### Table A–39 Fields in the Content Rule Dialog Box

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Structural Object Class</td>
<td>The name of the structural object class to which you want to assign this content rule</td>
</tr>
<tr>
<td>Object ID</td>
<td>The unique identifier of the content rule you are creating</td>
</tr>
<tr>
<td>Label</td>
<td>A descriptive friendly name of this content rule</td>
</tr>
<tr>
<td>Auxiliary Classes</td>
<td>The auxiliary object classes whose attributes you want to associate with the specified structural object class. To specify an auxiliary class:</td>
</tr>
<tr>
<td></td>
<td>1. Choose Add. The Auxiliary Class Selector dialog box appears.</td>
</tr>
<tr>
<td></td>
<td>2. Select the auxiliary class you want to add.</td>
</tr>
<tr>
<td></td>
<td>3. Choose Select. This returns you to the New Content Rule dialog box. The auxiliary class you just specified appears in the Auxiliary Classes field.</td>
</tr>
<tr>
<td>Mandatory Attributes</td>
<td>The mandatory attributes you want to associate with the specified structural object class. To specify a mandatory attribute:</td>
</tr>
<tr>
<td></td>
<td>2. Select the mandatory attribute you want to add. If you want this attribute indexed, then select the corresponding check box in the Indexed column.</td>
</tr>
<tr>
<td></td>
<td>3. Choose Select. This returns you to the New Content Rule dialog box. The mandatory attribute you just specified appears in the Mandatory Attributes field.</td>
</tr>
<tr>
<td>Optional Attributes</td>
<td>The optional attributes you want to associate with the specified structural object class. To specify an optional attribute:</td>
</tr>
<tr>
<td></td>
<td>1. Choose Add. The Optional Attribute Selector dialog box appears.</td>
</tr>
<tr>
<td></td>
<td>2. Select the optional attribute you want to add. If you want this attribute indexed, then select the corresponding check box in the Indexed column.</td>
</tr>
<tr>
<td></td>
<td>3. Choose Select. This returns you to the New Content Rule dialog box. The optional attribute you just specified appears in the Optional Attributes field.</td>
</tr>
</tbody>
</table>
Server Management Fields in Oracle Directory Manager

This section contains these topics:

- Configuration Sets Fields in Oracle Directory Manager
- System Operational Attributes Fields in Oracle Directory Manager
- Super, Guest, and Proxy User Fields in Oracle Directory Manager
- Query Optimization Fields in Oracle Directory Manager
- Entry Search Fields and Buttons in Oracle Directory Manager

Configuration Sets Fields in Oracle Directory Manager

See Also: Managing Server Configuration Set Entries by Using Oracle Directory Manager on page 7-3

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max. Number of DB Connections</td>
<td>Type the number of concurrent database connections a single directory server process can have. The default is ten.</td>
</tr>
<tr>
<td>Number of Child Processes</td>
<td>Type the number of server processes a single instance can spawn. The default is one.</td>
</tr>
<tr>
<td>Non-SSL Port</td>
<td>The default non-SSL port is 389. You can change the non-SSL port.</td>
</tr>
<tr>
<td>Set</td>
<td>The this field displays the number of the configuration set entry. The default configuration set is 0. There can be as many different configuration sets as needed. The same configuration set can be used by more than one instance if the parameter needs of the multiple instances are the same. The set number is not modifiable.</td>
</tr>
<tr>
<td>SASL Authentication Mode</td>
<td>The default value is 1. No other values are supported in this release of Oracle Internet Directory.</td>
</tr>
<tr>
<td>SASL Mechanism</td>
<td>The default value is DIGEST-MD5. No other values are supported in this release of Oracle Internet Directory.</td>
</tr>
<tr>
<td>SASL Cipher Choice</td>
<td>The default values for this multivalued attribute are:</td>
</tr>
<tr>
<td></td>
<td>RC4-56</td>
</tr>
<tr>
<td></td>
<td>DES</td>
</tr>
<tr>
<td></td>
<td>3DES</td>
</tr>
</tbody>
</table>

Table A–41 Fields in the Configuration Sets: SSL Settings Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSL Authentication</td>
<td>Choose one of the following:</td>
</tr>
<tr>
<td></td>
<td>No SSL Authentication: Neither the client nor the server authenticates itself to the other. No certificates are sent or exchanged. In this case, SSL encryption/decryption only is used.</td>
</tr>
<tr>
<td></td>
<td>SSL Client and Server Authentication: Both client and server authenticate themselves to each other and send certificates to each other.</td>
</tr>
<tr>
<td></td>
<td>SSL Server Authentication: Only the directory server authenticates itself to the client. The directory server sends the client a certificate verifying that the server is authentic.</td>
</tr>
</tbody>
</table>
SSL Enable
Choose one of the following:

- Both SSL and Non-SSL: Both non-secure operation and SSL authentication
- Non-SSL Only: Only non-secure operation; default port is 389, changeable in the SSL Port field
- SSL Only: Only SSL authentication; default port is 636, changeable in the SSL Port field

SSL Wallet URL
Type the location of the server-side SSL wallet. If you elect to change the location of the wallet, you must change this parameter. You must set the wallet location on both the client and the server. For example, on UNIX, you could set this parameter as follows:

file:/home/my_dir/my_wallet

On Microsoft Windows, you could set this parameter as follows:

file:C:\my_dir\my_wallet

SSL Port
The default SSL port is 636. You can change the SSL port.

---

**System Operational Attributes Fields in Oracle Directory Manager**

See Also: "Setting System Operational Attributes by Using Oracle Directory Manager" on page 7-7

**Table A–42** System Operation Attributes Displayed in Oracle Directory Manager

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Default Value</th>
<th>Modifiable?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow Anonymous Binds</td>
<td>Indicator of whether anonymous binds are allowed or not. If set to 1, then anonymous binds are allowed. If set to 0 (zero), then they are not allowed.</td>
<td>1</td>
<td>Yes</td>
</tr>
<tr>
<td>Alternate Server</td>
<td>When connectivity to the local server is lost, clients have the option of accessing one of the servers listed in this attribute. Specify other Oracle directory servers in the system that have the same set of naming contexts as that of the local server. The format is: <code>ldap://host_name:port_number</code></td>
<td>None</td>
<td>Yes</td>
</tr>
<tr>
<td>Configuration Set Location</td>
<td>DN of the entry holding the top of the naming context in this server</td>
<td><code>cn=subconfigsubentry</code></td>
<td>No</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
<td>Default Value</td>
<td>Modifiable?</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-----------------------------------------------------------------------------</td>
<td>---------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Critical Event Level</td>
<td>Specify critical events related to security and system resources that you want recorded. Please note that for events other than super user, proxy and replication login, the value of the orclStatsFlag attribute also must be set to 1 for enabling this feature. See Also: “Configuring Critical Events” on page 14-19 for a list of critical events that can be monitored.</td>
<td>0</td>
<td>Yes</td>
</tr>
<tr>
<td>DIP Repository</td>
<td>Used by the directory replication server, and indicates whether change logs are to be generated in the consumer node for the Oracle directory integration and provisioning server to consume.</td>
<td>FALSE</td>
<td>Yes</td>
</tr>
<tr>
<td>Directory Version</td>
<td>The version or release of Oracle Internet Directory that you are using</td>
<td>9.0.4.0.0</td>
<td>No</td>
</tr>
<tr>
<td>Enable Entry Cache</td>
<td>Specify whether entry caching, described in “Entry Caching” on page 3-8, is enabled. The value for enabled is 1; the value for disabled is 0.</td>
<td>1</td>
<td>Yes</td>
</tr>
<tr>
<td>Enable Group Cache</td>
<td>The cache of privilege groups and ACL groups in the directory server. Using this cache improves the performance of access control evaluation for users when privilege and ACP groups are used in ACL. Use the group cache when a privilege group membership does not change frequently. If a privilege group membership does change frequently, then it is best to turn off the group cache. This is because, in such a case, computing a group cache increases overhead.</td>
<td>1</td>
<td>Yes</td>
</tr>
<tr>
<td>Enable Match DN Processing</td>
<td>If the base DN of a search request is not found, then the directory server returns the nearest DN that matches the specified base DN. Whether the directory server tries to find the nearest match DN is controlled by this attribute. If set to 1, then match DN processing is enabled. If set to 0, then match DN processing is disabled.</td>
<td>1</td>
<td>Yes</td>
</tr>
<tr>
<td>Enable Statistics Gathering</td>
<td>Indicator of whether you want to enable or disable the Oracle Internet Directory Server Manageability framework. To enable, set this to 1. To disable, set it to 0.</td>
<td>0</td>
<td>Yes</td>
</tr>
<tr>
<td>Entry Cache Size in Bytes</td>
<td>The maximum number of bytes of RAM that the entry cache can use.</td>
<td>100M</td>
<td>Yes</td>
</tr>
</tbody>
</table>
### Table A–42  (Cont.) System Operation Attributes Displayed in Oracle Directory Manager

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Default Value</th>
<th>Modifiable?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Indexed Attribute Locations</td>
<td>Specify the DN for the file containing all indexed attributes</td>
<td>cn=catalogs</td>
<td>No</td>
</tr>
<tr>
<td>Maximum Entries in Entry Cache</td>
<td>Specify the maximum number of entries that can be present in the entry cache.</td>
<td>25,000</td>
<td>Yes</td>
</tr>
<tr>
<td>Maximum TCP</td>
<td>Specify how long the server should keep an idle connection open before closing it.</td>
<td>120</td>
<td></td>
</tr>
<tr>
<td>Connection Idle Time</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Naming Contexts</td>
<td>Specify the topmost DNs of naming contexts in this server that you want to publish. You must have super user privileges to publish a DN as a naming context.</td>
<td>None</td>
<td>Yes</td>
</tr>
<tr>
<td>Password Encryption</td>
<td>Hash algorithm for encrypting the password. Options are:</td>
<td>MD4</td>
<td>Yes</td>
</tr>
<tr>
<td></td>
<td>■ MD4 Secure Hash Algorithm</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ MD5 Secure Hash Algorithm</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ No encryption</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ SHA</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>■ UNIX Crypt</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Process Instance Location</td>
<td>DN of the entry holding the Instance Registry in this server</td>
<td>cn=subregistysubentry</td>
<td>No</td>
</tr>
<tr>
<td>Query Entry Return Limit</td>
<td>Maximum number of entries to be returned by a search</td>
<td>1000</td>
<td>Yes</td>
</tr>
<tr>
<td>Replica ID</td>
<td>Unique identifier of a node in a replication agreement</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Replication Agreements</td>
<td>DN of the entry holding the replication agreement</td>
<td>cn=orclareplagreements</td>
<td>No</td>
</tr>
<tr>
<td>Replication Log Location</td>
<td>DN of the entry holding the change log in this server</td>
<td>cn=changelog</td>
<td>No</td>
</tr>
<tr>
<td>Replication Status Location</td>
<td>DN of the entry holding the change status in this server</td>
<td>cn=changelogstatus</td>
<td>No</td>
</tr>
<tr>
<td>Schema Definition Location</td>
<td>DN of the schema</td>
<td>cn=subschemasubentry</td>
<td>No</td>
</tr>
<tr>
<td>Server Mode</td>
<td>Indicator of whether data can be written to the server. You can change this value to either read/write or read-only. Change the default to read-only during replication process.</td>
<td>read/write</td>
<td>Choices are Read/Write, Read/Modify and Read-Only</td>
</tr>
<tr>
<td>Server Operation Time Limit</td>
<td>Maximum amount of time, in seconds, allowed for a search to be completed</td>
<td>3600</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Server Management Fields in Oracle Directory Manager

---

### Table A–42 (Cont.) System Operation Attributes Displayed in Oracle Directory Manager

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
<th>Default Value</th>
<th>Modifiable?</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Simple Modify Changelog Attribute</strong></td>
<td>In a multimaster replication group, resolving conflicts for changes in some attribute values can require considerable resources. You can avoid this performance degradation by specifying those attributes in this field. When you specify attributes in this field, any changes to the values of those attributes are reflected in the change log. However, in a multimaster replication group, conflict resolution for those attributes is turned off.</td>
<td>uniquemember member</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Statistics Collection Interval</strong></td>
<td>Specify how often you want to gather sample statistics—that is, the number of minutes in the interval. Set this to 1 or more minutes.</td>
<td>60</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Statistics Level</strong></td>
<td>Specify whether you want to enable or disable the Oracle Internet Directory Server Manageability framework. To enable, set this to 1. To disable, set it to 0.</td>
<td>0</td>
<td>Yes</td>
</tr>
<tr>
<td><strong>Supported Control List</strong></td>
<td>Enter extension information for any LDAP operation. The control types supported by Oracle Internet Directory are listed as values of the <code>supportedcontrol</code> attribute in the root DSE. Each control type has an associated object identifier defined by the LDAP standard. The values of the <code>supportedcontrol</code> attribute are standard object identifiers assigned to control types.</td>
<td>manageDSACtrl</td>
<td>No</td>
</tr>
<tr>
<td><strong>Supported Extension</strong></td>
<td>The unique identifiers of proprietary extensions to LDAP operations that are supported in this release of Oracle Internet Directory. In Release 9.0.4, there is one extended operation. It enables a plug-in using a PL/SQL package in the database to bind to the directory server.</td>
<td>2.16.840.1.113894.1.9.1</td>
<td>No</td>
</tr>
<tr>
<td><strong>Supported LDAP Version</strong></td>
<td>LDAP version that Oracle Internet Directory supports</td>
<td>LDAP Version 2, LDAP Version 3</td>
<td>No</td>
</tr>
<tr>
<td><strong>Supported SASL Mechanisms</strong></td>
<td>Some clients can use the Simple Authentication and Security Layer (SASL). This field indicates the authentication mechanisms supported by the directory server. See Also: &quot;Authentication by Using Simple Authentication and Security Layer (SASL)&quot; on page 16-8</td>
<td>DIGEST-MD5</td>
<td>No</td>
</tr>
<tr>
<td><strong>Upgrade in Progress</strong></td>
<td>Reserved for upgrade</td>
<td>FALSE</td>
<td>No</td>
</tr>
</tbody>
</table>

---

A-28 Oracle Internet Directory Administrator's Guide
Super, Guest, and Proxy User Fields in Oracle Directory Manager

See Also: "Managing Super Users, Guest Users, and Proxy Users by Using Oracle Directory Manager" on page 7-9

Table A–43 Fields in the System Passwords Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Super User Name</td>
<td>Type the super user name, or choose Browse to search for it. The default is orcladmin.</td>
</tr>
<tr>
<td>Super User Password</td>
<td>Type the super user password. The default is the same as the password you specified for the Oracle Application Server administrator (ias_admin) during installation. You should change this password immediately.</td>
</tr>
<tr>
<td>Guest Login Name</td>
<td>Type the guest login name, or choose Browse to search for it. Guests have privileges determined by the access control list (ACL) in the directory. The default is guest.</td>
</tr>
<tr>
<td>Guest Login Password</td>
<td>Type the guest login password. The default is guest.</td>
</tr>
<tr>
<td>Proxy Login Name</td>
<td>Type the proxy login name, or choose Browse to search for it. Proxy users have privileges determined by the ACPs in the directory. The default is proxy.</td>
</tr>
<tr>
<td>Proxy Login Password</td>
<td>Type the proxy login password. The default is proxy. You should change this password immediately.</td>
</tr>
</tbody>
</table>

Query Optimization Fields in Oracle Directory Manager

See Also: "Optimizing Searches for Skewed Attributes by Using Oracle Directory Manager" on page 25-11

Table A–44 Fields in the Query Optimization Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Trigger Dynamic Group Cache Refresh</td>
<td>Select to force a refresh of the dynamic group cache.</td>
</tr>
<tr>
<td>Skip Referral Process</td>
<td>Select to skip referral in SQL generated for searches. If there are no referral entries in the directory; skipping referral will help optimizing search performance.</td>
</tr>
<tr>
<td>Force Flush Debug Messages</td>
<td>Select to enable debug messages to be written to the log file when a message is logged by the directory server. This feature is disabled by default.</td>
</tr>
<tr>
<td>Maximum Number of Cached Users Group Connection</td>
<td>Specifies the number of connection DNs whose privileged groups can be cached. The default value is 25000 identities (connection DNs). Increase the value if your installation has more than 25000 users.</td>
</tr>
<tr>
<td>Maximum Number of Cached Search Entries in BER</td>
<td>Specifies the maximum allowed BER entry. When searching a subtree, the server does not write to the client until this number of entries have been processed. By default this value is 5. If the entries are larger than 8000 bytes, then reduce this value to 1.</td>
</tr>
<tr>
<td>Maximum Time for OID Server to Read/Write to LDAP Client</td>
<td>Specifies, in seconds, the network read/write time out. When an LDAP client initiates an operation, then does not respond to the server for this number of seconds, the server closes the connection. The default is 300 seconds.</td>
</tr>
</tbody>
</table>
### Table A–44 (Cont.) Fields in the Query Optimization Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum Entry Size in Cache</td>
<td>Specifies, in bytes, the upper size limit of entries stored in the cache. The default is 5000—that is, 5 kilobytes.</td>
</tr>
<tr>
<td>Monitored Users’ DNs</td>
<td>Specifies the list of user DNs for which to track LDAP operations.</td>
</tr>
<tr>
<td>Level for Security Events Tracking</td>
<td>Specifies levels of bind and compare information collection. Possible levels are:</td>
</tr>
<tr>
<td></td>
<td>■ 1–Bind DN only</td>
</tr>
<tr>
<td></td>
<td>■ 2–Bind DN and IP address</td>
</tr>
<tr>
<td></td>
<td>■ 4–Compare DN only</td>
</tr>
<tr>
<td></td>
<td>■ 8–Compare DN and IP address</td>
</tr>
<tr>
<td></td>
<td>■ 16–Compare DN, IP address and failure details</td>
</tr>
<tr>
<td></td>
<td>Bind and Compare levels may be added. For example, 18 specifies Bind DN and IP address, plus Compare DN, IP address and failure details.</td>
</tr>
<tr>
<td>Maximum RAM Space Used to Track Security Events</td>
<td>Specifies the maximum memory, in bytes, that is used to track security events. The default is 100 megabytes.</td>
</tr>
<tr>
<td>PKI Mapping/Matching Rule</td>
<td>Specifies the matching rule for mapping a user’s PKI certificate DN to the user’s entry DN in Oracle Internet Directory. The following matching rule values are allowed:</td>
</tr>
<tr>
<td></td>
<td>■ 0–Exact match. The PKI certificate DN must match the user entry DN.</td>
</tr>
<tr>
<td></td>
<td>■ 1–Certificate hash search. Check to see if the user has a PKI certificate provisioned into Oracle Internet Directory.</td>
</tr>
<tr>
<td></td>
<td>■ 2–A combination of exact match and certificate hash search. If the exact match fails, then a certificate search is performed.</td>
</tr>
<tr>
<td></td>
<td>■ 3–Mapping rule only. Use a mapping rule to map user PKI certificate DNs to Oracle Internet Directory DNs.</td>
</tr>
<tr>
<td></td>
<td>■ 4–Try in order: 1 (mapping rule), 2 (certificate hash search), 3 (exact match).</td>
</tr>
<tr>
<td>Dump Flags</td>
<td>Dump Flags</td>
</tr>
<tr>
<td>LDAP Connection Timeout</td>
<td>Enter the maximum number of seconds that the directory client can remain idle before terminating the connection. The default is 0, meaning that there is no timeout.</td>
</tr>
<tr>
<td>Time Limit Mode</td>
<td>To adjust server performance, set the search time limit to be either accurate or approximate. If you specify it as accurate, then searches end precisely at the specified number of seconds. If you specify it as approximate, then searches end within a few seconds of the specified number of seconds. In smaller workloads, specifying it as approximate provides better performance.</td>
</tr>
<tr>
<td>Attributes with Low Cardinality</td>
<td>Enter the attributes you want to designate as skewed. See Also: “Optimizing Searches” on page 25-9 for a discussion of skewed attributes.</td>
</tr>
</tbody>
</table>
Entry Search Fields and Buttons in Oracle Directory Manager

**Table A–45  Search Filters for Entries**

<table>
<thead>
<tr>
<th>Filter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Begins With</td>
<td>Searches by using only the first few characters of the attribute's value. For example, cn Begins With Fran retrieves all entries in which the first few letters of the cn attribute are Fran. These would include Frank, Fran, Frances, Franklin, and so on.</td>
</tr>
<tr>
<td>Ends With</td>
<td>Searches for an entry by using only the last few characters of the specified attribute's value. For example, cn Ends With son retrieves Baldisson, Jacobson, Johnson, and so on.</td>
</tr>
<tr>
<td>Contains</td>
<td>Searches for an entry in which the attribute you specified includes, but is not necessarily limited to, the value you enter. For example, cn Contains Wins retrieves all entries in which the cn attribute contains the letters wins. These would include Winslow, Czerwinski, Winship, and so on.</td>
</tr>
<tr>
<td>Exact Match</td>
<td>Searches for an entry whose specified attribute is the same as the value you enter. For example, cn Exactly Matches Franklin Baldwins retrieves all entries with the cn attribute has the value Franklin Baldwins.</td>
</tr>
<tr>
<td>Greater or Equal</td>
<td>Searches for an entry in which the specified attribute is numerically or alphabetically greater than or equal to the value you enter. For example, cn Greater or Equal Frank retrieves all entries with cn attributes that range from the first Frank to the end of the alphabet.</td>
</tr>
<tr>
<td>Less or Equal</td>
<td>Searches for entries in which the specified attribute is numerically or alphabetically less than or equal to the value you enter. For example, cn Less or Equal Frank retrieves all cn attributes from the first Frank to the beginning of the alphabet.</td>
</tr>
<tr>
<td>Present</td>
<td>Determines if an entry with the specified attribute is present at that level of the tree. You do not need to enter a value to use this relationship. The phrase cn Present retrieves all entries with the cn attribute at that level of the tree.</td>
</tr>
</tbody>
</table>

**Table A–46  Buttons Used in Searches for Entries**

<table>
<thead>
<tr>
<th>Button</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>New</td>
<td>Creates a new search criteria bar in the Criteria field. This button is enabled only when the Criteria field is empty.</td>
</tr>
<tr>
<td>And</td>
<td>Creates another search criteria bar in the Criteria field. Matches all entries with one specified attribute with those that also have another specified attribute. For example, cn=Baldwins And title=Laborer retrieves all Baldwins who are also laborers.</td>
</tr>
<tr>
<td>Or</td>
<td>Creates another search criteria bar in the Criteria field. Matches all entries with either one specified attribute or another. For example, title=Laborer Or title=Foreman retrieves all employees who are either laborers or foremen.</td>
</tr>
<tr>
<td>Not</td>
<td>Negates the criterion in the selected search criteria bar and retrieves all entries that do not have the specified criterion. For example, cn=Frank And Not title=Laborer retrieves all persons named Frank who are not laborers.</td>
</tr>
<tr>
<td>Delete</td>
<td>Deletes a selected search criteria bar</td>
</tr>
</tbody>
</table>

**Server Management Fields in Oracle Directory Manager**
SSL Management Fields in Oracle Directory Manager

See Also:
- Table A-41, "Fields in the Configuration Sets: SSL Settings Tab Page" on page A-24
- "Configuring SSL Parameters by Using Oracle Directory Manager" on page 17-3

Table A–47 Fields in the SSL Settings Tab Page

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSL Authentication</td>
<td>Choose one of the following:</td>
</tr>
<tr>
<td>- 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.</td>
<td></td>
</tr>
<tr>
<td>- SSL Client and Server Authentication: Two-way authentication. Both client and server send certificates to each other.</td>
<td></td>
</tr>
<tr>
<td>- SSL Server Authentication: One-way authentication. Only the directory server authenticates itself to the client by sending its certificate to the client.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SSL Enable</th>
<th>Choose one of the following:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Both SSL and non-SSL: for both non-secure operation and SSL authentication</td>
<td></td>
</tr>
<tr>
<td>- Non-SSL only: for non-secure operation only</td>
<td></td>
</tr>
<tr>
<td>- SSL only: for SSL authentication only</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SSL Wallet URL</th>
<th>Type the location of the server-side SSL wallet. If you elect to change the location of the wallet, you must change this parameter. You must set the wallet location on both the client and the server. For example, on UNIX, you could set this parameter as follows:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>file:/home/my_dir/my_wallet</td>
</tr>
<tr>
<td></td>
<td>On Microsoft Windows, you could set this parameter as follows:</td>
</tr>
<tr>
<td></td>
<td>file:C:\my_dir\my_wallet</td>
</tr>
</tbody>
</table>

| SSL Port           | The default SSL port is 636. You can change the SSL port.                   |
Synchronization Fields in Oracle Directory Manager

This section describes the fields in Oracle Directory Manager for administering directory synchronization. These are fields for registering a directory integration profile.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Profile Name</td>
<td>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.</td>
</tr>
<tr>
<td>Synchronization Mode</td>
<td>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.</td>
</tr>
<tr>
<td>Profile Status</td>
<td>Specify whether the profile is enabled or disabled. This field is mandatory. The default is ENABLE.</td>
</tr>
<tr>
<td>Profile Password</td>
<td>Specify the password that directory integration server is to use when binding to Oracle Internet Directory on behalf of the profile. This field is mandatory and the default is welcome.</td>
</tr>
<tr>
<td>Scheduling Interval</td>
<td>Specify the number of seconds between synchronization attempts between a connected directory and Oracle Internet Directory. This field is mandatory. The default is 60.</td>
</tr>
<tr>
<td>Maximum Number of Retries</td>
<td>Specify the maximum number of times the directory integration 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.</td>
</tr>
<tr>
<td>Profile Version</td>
<td>Version of Oracle Directory Integration Platform with which this profile was created.</td>
</tr>
</tbody>
</table>
Table A–49  Fields on the Execution Tab for Synchronization in Oracle Directory Manager

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent Execution Command</td>
<td>Specify the agent executable name and the arguments used by the directory integration 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.</td>
</tr>
<tr>
<td>Connected Directory Account</td>
<td>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.</td>
</tr>
<tr>
<td>Connected Directory Account Password</td>
<td>Specify the password the connector/agent is to use when accessing the connected directory. This field is optional. There is no default.</td>
</tr>
<tr>
<td>Additional Config Info</td>
<td>This field displays additional information that the directory integration server passes to an agent. You cannot modify this field through Oracle Directory Manager. The only way to modify it is to use ldapuploadagentfile.sh. There is no default.</td>
</tr>
<tr>
<td>Connected Directory URL</td>
<td>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. <strong>Note:</strong> To connect to SunONE Directory Server by using SSL, the server certificate needs to be loaded into the wallet. <strong>See Also:</strong> The chapter on Oracle Wallet Manager in <em>Oracle Advanced Security Administrator’s Guide</em></td>
</tr>
<tr>
<td>Interface Type</td>
<td>The format used by the import or export file. Options are DB, LDAP, LDIF, and TAGGED. This field is optional. The default is TAGGED.</td>
</tr>
</tbody>
</table>
Table A–50  Fields on the Mapping Tab Page for Synchronization in Oracle Directory Manager

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mapping Rules</td>
<td>This field displays the mapping rules for converting data between a connected directory and Oracle Internet Directory. There is no default.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> You cannot edit the mapping rules file by using Oracle Directory Manager. You edit the mapping rules in a file manually and then upload it to the profile by using <code>dipassistant</code>. See the <code>dipassistant</code> command-line tool reference in Oracle Identity Management User Reference.</td>
</tr>
<tr>
<td>Connected Directory</td>
<td>Specify the attribute that uniquely identifies an entry in the connected directory.</td>
</tr>
<tr>
<td>Matching Filter</td>
<td></td>
</tr>
<tr>
<td>OID Matching Filter</td>
<td>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.</td>
</tr>
</tbody>
</table>

Table A–51  Fields on the Status Tab Page for Synchronization in Oracle Directory Manager

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OID Last Applied Change Number (Import operations only)</td>
<td>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 conscientiously 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.</td>
</tr>
<tr>
<td>Last Execution Time</td>
<td>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.</td>
</tr>
<tr>
<td>Last Successful Execution Time</td>
<td>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.</td>
</tr>
<tr>
<td>Synchronization Status</td>
<td>Synchronization success/failure.</td>
</tr>
<tr>
<td>Synchronization Errors</td>
<td>The last error message. You cannot modify this field. There is no default.</td>
</tr>
<tr>
<td>Last Applied Change Number (Export operations only)</td>
<td>The number of the change log entry that was most recently applied successfully to the connected directory. The field can be conscientiously 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.</td>
</tr>
</tbody>
</table>

Server Chaining Management

Table A–52  Fields on the Server Chaining Management Window (for Active Directory or iPlanet)

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Authentication</td>
<td>Enable external authentication capability.</td>
</tr>
<tr>
<td>Enable Modification</td>
<td>Enable external modification capability.</td>
</tr>
<tr>
<td>Field</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Enable Search</td>
<td>Enable external search capability.</td>
</tr>
<tr>
<td>User Container</td>
<td>The user container in the external directory from which to perform the user search operation.</td>
</tr>
<tr>
<td>Target User Container</td>
<td>The user container in Oracle Internet Directory in which the external users reside.</td>
</tr>
<tr>
<td>Group Container</td>
<td>The group container in the external directory from which to perform the group search operation.</td>
</tr>
<tr>
<td>Target Group Container</td>
<td>The group container in the external directory from which to perform the group search operation.</td>
</tr>
<tr>
<td>Host</td>
<td>The host name of the external directory host. This is a single value attribute.</td>
</tr>
<tr>
<td>Port</td>
<td>The port number of the external directory host. The default value is 389.</td>
</tr>
<tr>
<td>Login User DN</td>
<td>The DN in the external directory. Server chaining will bind against the external directory using this identity to perform search and modify operations. This identity must have sufficient privilege to perform the operation.</td>
</tr>
<tr>
<td>Login User Password</td>
<td>The password for the DN of the external directory.</td>
</tr>
<tr>
<td>Attribute Mapping</td>
<td>Specifies each attribute mapping between the external directory and Oracle Internet Directory. For example, to map the eMail attribute from the target directory to the mail attribute in Oracle Internet Directory, set this attribute to: OID Attribute: mail, Target Directory Attribute: eMail</td>
</tr>
</tbody>
</table>

Table A–52 (Cont.) Fields on the Server Chaining Management Window (for Active Directory or iPlanet)
The paper contained in this appendix is copied with permission from RFC 2254 of the Internet Engineering Task Force. The URL for this document is:
http://www.ietf.org

The contents of this paper may have been superseded by later papers or other information. Check the above Web site and related sites for additional or supplementary information.

NOTE: ORACLE DISCLAIMS ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Network Working Group                                          T. Howes
Request for Comments: 2254                Netscape Communications Corp.
Category: Standards Track                                 December 1997

The String Representation of LDAP Search Filters

1. Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

Copyright Notice

Copyright (C) The Internet Society (1997). All Rights Reserved.

IESG Note

This document describes a directory access protocol that provides both read and update access. Update access requires secure authentication, but this document does not mandate implementation of any satisfactory authentication mechanisms.

In accordance with RFC 2026, section 4.4.1, this specification is being approved by IESG as a Proposed Standard despite this limitation, for the following reasons:

a. to encourage implementation and interoperability testing of these protocols (with or without update access) before they are deployed, and
b. to encourage deployment and use of these protocols in read-only applications. (e.g. applications where LDAPv3 is used as a query language for directories which are updated by some secure mechanism other than LDAP), and

c. to avoid delaying the advancement and deployment of other Internet standards-track protocols which require the ability to query, but not update, LDAPv3 directory servers.

Readers are hereby warned that until mandatory authentication mechanisms are standardized, clients and servers written according to this specification which make use of update functionality are UNLIKELY TO INTEROPERATE, or MAY INTEROPERATE ONLY IF AUTHENTICATION IS REDUCED TO AN UNACCEPTABLY WEAK LEVEL.

Implementors are hereby discouraged from deploying LDAPv3 clients or servers which implement the update functionality, until a Proposed Standard for mandatory authentication in LDAPv3 has been approved and published as an RFC.

2. Abstract

The Lightweight Directory Access Protocol (LDAP) [1] defines a network representation of a search filter transmitted to an LDAP server. Some applications may find it useful to have a common way of representing these search filters in a human-readable form. This document defines a human-readable string format for representing LDAP search filters.

This document replaces RFC 1960, extending the string LDAP filter definition to include support for LDAP version 3 extended match filters, and including support for representing the full range of possible LDAP search filters.

3. LDAP Search Filter Definition

An LDAPv3 search filter is defined in Section 4.5.1 of [1] as follows:

\[
\text{Filter} ::= \text{CHOICE} \{
\text{and} \quad [0] \text{SET OF Filter},
\text{or} \quad [1] \text{SET OF Filter},
\text{not} \quad [2] \text{Filter},
\text{equalityMatch} \quad [3] \text{AttributeValueAssertion},
\text{substrings} \quad [4] \text{SubstringFilter},
\text{greaterOrEqual} \quad [5] \text{AttributeValueAssertion},
\text{lessOrEqual} \quad [6] \text{AttributeValueAssertion},
\text{present} \quad [7] \text{AttributeDescription},
\text{approxMatch} \quad [8] \text{AttributeValueAssertion},
\text{extensibleMatch} \quad [9] \text{MatchingRuleAssertion}
\}\n\]

\[
\text{SubstringFilter} ::= \text{SEQUENCE} \{
\text{type} \quad \text{AttributeDescription},
\text{SEQUENCE OF CHOICE} \{
\text{initial} \quad [0] \text{LDAPString},
\}
\}\n\]
The LDAP Filter Definition

any [1] LDAPString,

final [2] LDAPString

}

AttributeValueAssertion ::= SEQUENCE {
    attributeDesc AttributeDescription,
    attributeValue AttributeValue
}

MatchingRuleAssertion ::= SEQUENCE {
    matchingRule [1] MatchingRuleID OPTIONAL,
    type [2] AttributeDescription OPTIONAL,
    matchValue [3] AssertionValue,
    dnAttributes [4] BOOLEAN DEFAULT FALSE
}

AttributeDescription ::= LDAPString

AttributeValue ::= OCTET STRING

MatchingRuleID ::= LDAPString

AssertionValue ::= OCTET STRING

LDAPString ::= OCTET STRING

where the LDAPString above is limited to the UTF-8 encoding of the ISO 10646 character set [4]. The AttributeDescription is a string representation of the attribute description and is defined in [1].

The AttributeValue and AssertionValue OCTET STRING have the form defined in [2]. The Filter is encoded for transmission over a network using the Basic Encoding Rules defined in [3], with simplifications described in [1].

4. String Search Filter Definition

The string representation of an LDAP search filter is defined by the following grammar, following the ABNF notation defined in [5]. The filter format uses a prefix notation.

filter = "(" filtercomp ")"

filtercomp = and / or / not / item

and = ";" filterlist

or = "|" filterlist

not = "!" filter

filterlist = 1*filter

item = simple / present / substring / extensible

simple = attr filtertype value

filtertype = equal / approx / greater / less

equal = "="
approx = "~=
greater = ">=
less = "<="

extensible = attr [:dn] [: matchingrule] "=": value
          / [:dn] ":": matchingrule "=": value

present = attr "="
substring = attr '=' [initial] any [final]
initial = value
any = "*" *(value "*"
final = value

attr = AttributeDescription from Section 4.1.5 of [1]
matchingrule = MatchingRuleId from Section 4.1.9 of [1]
value = AttributeValue from Section 4.1.6 of [1]

The attr, matchingrule, and value constructs are as described in the corresponding section of [1] given above.

If a value should contain any of the following characters

<table>
<thead>
<tr>
<th>Character</th>
<th>ASCII value</th>
</tr>
</thead>
<tbody>
<tr>
<td>*</td>
<td>0x2a</td>
</tr>
<tr>
<td>(</td>
<td>0x28</td>
</tr>
<tr>
<td>)</td>
<td>0x29</td>
</tr>
<tr>
<td>\</td>
<td>0x5c</td>
</tr>
<tr>
<td>NUL</td>
<td>0x00</td>
</tr>
</tbody>
</table>

then the character must be encoded as the backslash '\\' character (ASCII 0x5c) followed by the two hexadecimal digits representing the ASCII value of the encoded character. The case of the two hexadecimal digits is not significant.

This simple escaping mechanism eliminates filter-parsing ambiguities and allows any filter that can be represented in LDAP to be represented as a NUL-terminated string. Other characters besides the ones listed above may be escaped using this mechanism, for example, non-printing characters.

For example, the filter checking whether the "cn" attribute contained a value with the character "*" anywhere in it would be represented as

"(cn=*\2a*)".

Note that although both the substring and present productions in the grammar above can produce the "attr=" construct, this construct is used only to denote a presence filter.

5. Examples

This section gives a few examples of search filters written using this notation.

(cn=Babs Jensen)
(!(!(cn=Tim Howes))
(& (objectClass=Person) ( (sn=Jensen) (cn=Babs J*)) )

(o=univ*of*mich*)

The following examples illustrate the use of extensible matching.

(cn:1.2.3.4.5:=Fred Flintstone)
(sn:dn:2.4.6.8.10:=Barney Rubble)
(o:dn:=Ace Industry)
(:,dn:2.4.6.8.10:=Dino)

The second example illustrates the use of the ":dn" notation to indicate that matching rule "2.4.6.8.10" should be used when making comparisons, and that the attributes of an entry’s distinguished name should be considered part of the entry when evaluating the match.

The third example denotes an equality match, except that DN components should be considered part of the entry when doing the match.

The fourth example is a filter that should be applied to any attribute supporting the matching rule given (since the attr has been left off). Attributes supporting the matching rule contained in the DN should also be considered.

The following examples illustrate the use of the escaping mechanism.

(o=Parens R Us \28for all your parenthetical needs\29)
(cn=\\2A*)
(filename=C:\5cMyFile)
(bin=\00\00\00\04)
(sn=Lu\c4\8di\c4\87)

The first example shows the use of the escaping mechanism to represent parenthesis characters. The second shows how to represent a "*" in a value, preventing it from being interpreted as a substring indicator. The third illustrates the escaping of the backslash character.

The fourth example shows a filter searching for the four-byte value 0x00000004, illustrating the use of the escaping mechanism to represent arbitrary data, including NUL characters.

The final example illustrates the use of the escaping mechanism to represent various non-ASCII UTF-8 characters.

6. Security Considerations

This memo describes a string representation of LDAP search filters. While the representation itself has no known security implications, LDAP search filters do. They are interpreted by LDAP servers to select entries from which data is retrieved. LDAP servers should take care to protect the data they maintain from unauthorized access.

7. References


[3] Specification of ASN.1 encoding rules: Basic, Canonical, and
[4] Yergeau, F., "UTF-8, a transformation format of Unicode and ISO
   10646", RFC 2044, October 1996.

8. Author’s Address
   Tim Howes
   Netscape Communications Corp.
   501 E. Middlefield Road
   Mountain View, CA 94043
   USA
   Phone: +1 415 937-3419
   EMail: howes@netscape.com

9. Full Copyright Statement
   Copyright (C) The Internet Society (1997). All Rights Reserved.
   This document and translations of it may be copied and furnished to others, and
derivative works that comment on or otherwise explain it or assist in its
implementation may be prepared, copied, published and distributed, in whole or in
part, without restriction of any kind, provided that the above copyright notice and this
paragraph are included on all such copies and derivative works. However, this
document itself may not be modified in any way, such as by removing the copyright
notice or references to the Internet Society or other
   Internet organizations, except as needed for the purpose of developing Internet
standards in which case the procedures for copyrights defined in the Internet
Standards process must be followed, or as required to translate it into languages other
than English.
   The limited permissions granted above are perpetual and will not be revoked by the
Internet Society or its successors or assigns.
   This document and the information contained herein is provided on an "AS IS" basis
and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE
DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT
LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN
WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF
MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.
The Access Control Directive Format

This appendix describes the format (syntax) of any access control item (ACI). It contains these topics:

- Schema for orclACI
- Schema for orclEntryLevelACI

**Schema for orclACI**

The access control directive defined by the user attribute orclACI has the following schema:

```
OrclACI:
{ object_identifier NAME 'orclACI' DESC 'Stores an inheritable ACI' EQUALITY
  accessDirectiveMatch SYNTAX 'accessDirectiveDescription' USAGE
  'directoryOperation'}
```

accessDirectiveDescription has the following BNF:

```
<accessDirectiveDescription>
::= access to <object> [by <subject> { <accessList> }]
```

```
<object> ::= [attr <EQ-OR-NEQ> ( * | (<attrList>) ) | entry]
```

```
<subject> ::= <entity> [<BindMode>] [Added_object_constraint=(<ldapFilter>)]
```

```
<entity> ::= * | self | dn="regex" | dnAttr={(dn_attribute)} | group="dn" |
guidattr={(guid_attribute)} | groupattr={(group_attribute)} | [SuperUser]
```

```
BindMode=(LDAP_authentication_choice) | LDAP_security_choice)
LDAP_authentication_choice:= proxy | simple | MD5Digest | PKCS12
LDAP_security_choice:= SSLNoAuth | SSLOneWay | SASL
```

```
<accessList> ::= <access> | <access>, <accessList>
```

```
<access> ::= none | compare | search | browse | proxy | read | selfwrite | write |
add | delete | nocompare | nosearch | nobrowse | noproxy | noread | noselfwrite |
nowrite | noadd | nodelete
```

```
<attrList> ::= <attribute name> | <attribute name>,<attrList>
```

```
<EQ-OR-NEQ> ::= = | !=
```

```
<regex> ::= <dn> | *,<dn_of_any_subtree_root>
```
Schema for orclEntryLevelACI

The entry level access control directive defined by the user attribute orclEntryLevelACI has the following schema:

"orclEntryLevelACI":
{ object_identifier NAME 'orclEntryLevelACI' DESC 'Stores entry level ACL Directive'
EQUALITY accessDirectiveMatch SYNTAX 'orclEntryLevelACIDescription'
USAGE 'directoryOperation' }

<orclEntryLevelACIDescription>
::= access to <object> [by <subject> ( <accessList> )]+

Note: The regular expression defined earlier is not meant to match any arbitrary expression. The syntax only allows expressions where the wild card is followed by a comma and a valid DN. The latter DN denoted by <dn_of_any_subtree_root> is intended to specify the root of some subtree.
Globalization Support in the Directory

Oracle Internet Directory uses Globalization Support to store, process and retrieve data in native languages. It ensures that Oracle Internet Directory utilities and error messages automatically adapt to the native language and locale.

This chapter discusses Globalization Support as used by Oracle Internet Directory and tells you the required NLS_LANG environment variables for the various components and tools in an Oracle Internet Directory environment.

See Also: “Globalization Support” on page 3-15 prior to configuring Globalization Support

This chapter contains these topics:

- About Character Sets and the Directory
- The NLS_LANG Environment Variable
- Using Non-AL32UTF8 Databases
- Using Globalization Support with LDIF Files
- Using Globalization Support with Command-Line LDAP Tools
- Setting NLS_LANG in the Client Environment
- Using Globalization Support with Bulk Tools

About Character Sets and the Directory

When computer systems process characters, they use numeric codes instead of the graphical representation of the character. For example, when the database stores the letter A, it actually stores a numeric code that is interpreted by software as the letter.

A group of characters (for example, alphabetic characters, ideographs, symbols, punctuation marks, and control characters) can be encoded as a character set. Each encoded character set assigns a unique code to each character in the set. For example, in the ASCII encoding scheme, the character code of the first character of the English upper-case alphabet is 0x4; in the EBCDIC encoding scheme, it is 0xc1.

The computer industry uses many encoded character sets. These character sets can differ in the number and types of characters available and in many other ways as well.

When you create a database, you specify an encoded character set. Choosing a character set determines, among other things, what languages can be represented in the database.
Oracle supports most national, international, and vendor-specific encoded character set standards.

This section contains the following topics:

- About Unicode
- About Oracle and UTF-8
- Migration from UTF8 to AL32UTF8 when Upgrading Oracle Internet Directory

About Unicode

No single character set contains enough characters to meet the requirements of day-to-day e-business requirements. For example, no one national character set can represent all the languages in the European Union. Moreover, there are potential conflicts between character sets because the same character can be represented by different codes in different character sets.

To overcome these obstacles, a global character set, called Unicode, was developed. It is a universal encoded character set that can store information from any language including punctuation marks, diacritics, mathematical symbols, technical symbols, musical symbols, and so forth. As of version 3.2, the Unicode Standard supports over 95,000 characters from the world’s alphabets, ideograph sets, and symbol collections. It includes 45,000 supplementary characters, most of which are Chinese, Japanese, and Korean characters that are rarely used but nevertheless need representation in electronic documentation.

Unicode has more than one implementation standard, and these are described in Table D–1.

<table>
<thead>
<tr>
<th>Implementation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>UTF-8</td>
<td>A variable-width 8-bit encoding of Unicode. One Unicode character can be one, two, three, or four bytes. Characters from European scripts are represented in one or two bytes. Those from Asian scripts are represented in three bytes, and supplementary characters are represented in four.</td>
</tr>
<tr>
<td>UCS-2</td>
<td>A fixed-width 16-bit encoding of Unicode in which each character, regardless of the script, is two bytes.</td>
</tr>
<tr>
<td>UTF-16</td>
<td>The 16-bit encoding of Unicode. It is an extension of UCS-2 that supports the supplementary characters added in Unicode 3.1. One character can be two or four bytes. Characters from European and Asian scripts are represented in two bytes, and supplementary characters are represented in four.</td>
</tr>
</tbody>
</table>

About Oracle and UTF-8

Oracle began supporting Unicode as a database character set beginning with Oracle database version 7. With Oracle9i, Oracle added a new UTF-8 character set called AL32UTF8. This database character set supports the latest version of Unicode (3.2), including the latest supplementary characters. Oracle intends to enhance AL32UTF8 as necessary to support future versions of the Unicode standard.

Migration from UTF8 to AL32UTF8 when Upgrading Oracle Internet Directory

Oracle Internet Directory now supports AL32UTF8. If you have upgraded Oracle Internet Directory from a version prior to 10g (10.1.4.0.1), then, for better performance,
Oracle recommends that you change the character set for the directory database from UTF8 to AL32UTF8. To do this:

1. Run the character set scanner (CSSCAN) to ensure that there are no invalid UTF8 characters inside your current database.
2. Run the CSALTER script to update the database to AL32UTF8.

See Also: The chapter on character set migration in the Oracle Database Globalization Support Guide in the Oracle Database Documentation Library

The NLS_LANG Environment Variable

The NLS_LANG parameter has three components—language, territory, and charset—in the form:

\[ \text{NLS\_LANG} = \text{language\_territory}.\text{charset} \]

Each component controls the operation of a subset of Globalization Support features. Components of the NLS_LANG parameter are shown in Table D–2.

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>language</td>
<td>Specifies conventions such as the language used for Oracle messages, day names, and month names. Each supported language has a unique name—for example, American English, French, or German. If language is not specified, the value defaults to American English. See Also: Oracle Database Globalization Support Guide in the Oracle Database Documentation Library for a complete list of languages</td>
</tr>
<tr>
<td>territory</td>
<td>Specifies conventions such as the default calendar, collation, date, monetary, and numeric formats. Each supported territory has a unique name; for example, America, France, or Canada. If territory is not specified, the value defaults to America. See Also: Oracle Database Globalization Support Guide in the Oracle Database Documentation Library for a complete list of territories</td>
</tr>
<tr>
<td>charset</td>
<td>Specifies the character set used by the client application (normally that of the user’s terminal). Each supported character set has a unique acronym, for example, WE8MSWIN1252, JA16SJIS, or AL32UTF8. See Also: Oracle Database Globalization Support Guide in the Oracle Database Documentation Library for a complete list of character sets</td>
</tr>
</tbody>
</table>

You can set NLS_LANG as an environment variable at the command line. The following are examples of legal values for NLS_LANG:

- AMERICAN_AMERICA.AL32UTF8
- JAPANESE_JAPAN.AL32UTF8
Using Non-AL32UTF8 Databases

You can run the Oracle directory server and database tools on a non-AL32UTF8 database, but be sure that all characters in the client character set are included in the database character set (with the same or different codes). Otherwise, you can lose data during ldapadd, ldapdelete, ldapmodify, or ldapmodifydn operations. For example, suppose that you perform an ldapadd operation using a multibyte character set on an underlying database that uses only single-byte characters. You will lose data because not all of the bytes you enter will be accepted by the database.

Using Globalization Support with LDIF Files

Attribute types are always ASCII strings that cannot contain multibyte characters. Oracle Internet Directory does not support multibyte characters in attribute type names. However, Oracle Internet Directory does support attribute values containing multibyte characters such as those in the simplified Chinese (ZHS16GBK) character set.

Attribute values can be encoded in different ways to allow Oracle Internet Directory tools to interpret them properly. There are two scenarios:

- An LDIF file Containing Only ASCII Strings
- An LDIF file Containing UTF-8 Encoded Strings

An LDIF file Containing Only ASCII Strings

In this scenario, character strings for attribute values are also in ASCII.

Because all tools use the UTF-8 character set by default, and ASCII is a proper subset of UTF-8, all tools can interpret these files. The same is true of keyboard input of values that are simply ASCII strings.

An LDIF file Containing UTF-8 Encoded Strings

In this scenario, character strings for attribute values are also in UTF-8.

Because, by default, all tools use the UTF-8 character set, all tools can interpret these files. The same is true of keyboard input of values that are UTF-8 strings.

In such a file, some characters may be multibyte. Multibyte characters strings can be present in the LDIF files as attribute values or given as keyboard input. They can be encoded in their native character set or in UTF-8. They can also be BASE64 encoded representations of either the native or the UTF-8 string.

Consider the following cases:

- CASE 1: Native Strings (Non-UTF-8)
- CASE 2: UTF-8 Strings
- CASE 3: BASE64 Encoded UTF-8 Strings
- CASE 4: BASE64 Encoded Native Strings

Because the directory server understands and expects only UTF-8 encoded strings, cases 1, 3, and 4 need to undergo conversion to UTF-8 strings before they can be sent to the LDAP server.
Using Globalization Support with Command-Line LDAP Tools

CASE 1: Native Strings (Non-UTF-8)
Use the -E character_set argument with the command-line tool dipassistant and with the command-line LDAP tools ldapadd, ldapaddmt, ldapbind, ldapcompare, ldapmoddn, ldapmod, ldapdelete, and ldapsearch. Use the encode="character_set" argument with the command-line bulk tools bulkload, bulkmodify, bulkdelete, and ldifwrite.

This example converts simplified Chinese native strings to UTF-8. The baseDN can be a simplified Chinese string:

```
ldapsearch -h my_host -p 389 -E ".ZHS16GBK" -b base_DN -s base "objectclass=*"
```

CASE 2: UTF-8 Strings
No conversion is required.

CASE 3: BASE64 Encoded UTF-8 Strings
You do not need to use the -E character_set argument with the command-line tools. Oracle Internet Directory tools automatically decode BASE64 encoded UTF-8 strings to UTF-8 strings.

CASE 4: BASE64 Encoded Native Strings
Use the -E character_set argument with the command-line tools dipassistant, ldapadd, ldapaddmt, ldapbind, ldapcompare, ldapmoddn, ldapmod, ldapdelete, and ldapsearch. Use the encode="character_set" argument with bulkload, bulkmodify, bulkdelete, and ldifwrite.

Oracle Internet Directory tools automatically decode BASE64 encoded native strings to simple native strings. The native strings are then converted to the equivalent UTF-8 strings.

Note: In any given input file, only one character set may be used.

Using Globalization Support with Command-Line LDAP Tools
The Oracle Internet Directory command-line tools read keyboard input or LDIF file input in the following ways:

- ASCII characters only
- Non-ASCII input (native language character set)
- BASE64 encoded values of UTF-8 or native strings (from LDIF file only)

If the character set being given as input from an LDIF file or keyboard is not UTF-8, then the command-line tools need to convert the input into UTF-8 format before sending it to the LDAP server.

You enable the command-line tools to convert the input into UTF-8 by specifying the -E character_set argument with dipassistant or any of the command-line LDAP tools. Use the encode="character_set" argument with bulkload, bulkmodify, bulkdelete, and ldifwrite.

This section contains these topics:

- Specifying the -E Argument When Using Each Tool
- Examples: Using the -E Argument with Command-Line LDAP Tools
Using Globalization Support with Command-Line LDAP Tools

Specifying the -E Argument When Using Each Tool

The client tools always assume UTF-8 (the Oracle character set name is AL32UTF8) to be the character set unless otherwise specified by the -E argument. The BASE64-encoded values are decoded, and then the decoded buffer is converted to UTF-8 if the -E argument is specified. For example, if you specify -E "\ZHS16GBK", then the decoded buffer is converted from simplified Chinese GBK to Unicode UTF-8 before being sent to the directory server.

Specifying the -E argument ensures that proper character set conversion can occur from the character set you specify for the -E argument (-E "character_set") to the AL32UTF8 character set.

The command-line tools use the -E argument to process the input in the character set specified for the -E argument. They display their output in the character set specified in the NLS_LANG environment variable.

For example, to add entries from an LDIF file encoded in the simplified Chinese character set (ZHS16GBK) by using ldapadd, type:

```bash
ldapadd -h myhost -p 389 -E \ZHS16GBK\ -f my_ldif_file
```

In this example, the ldapadd tool converts the characters from \\ZHS16GBK (simplified Chinese character set) to AL32UTF8” before they are sent across the wire to the directory server.

Examples: Using the -E Argument with Command-Line LDAP Tools

Table D–3 provides additional examples of how to use the -E argument correctly for each command-line tool. In each example, the command converts data from simplified Chinese, as specified by the value \ZHS16GBK, to AL32UTF8. For example, in each command, the values for the -D and -w options are in GBK. Specifying the -E argument converts them to UTF-8.

Note that, in the examples in Table D–3, we do not show any actual characters belonging to the \ZHS16GBK character set. These examples would, therefore, work without the -E argument. However, if the argument values contained actual characters in the \ZHS16GBK character set, then we would need to use the -E argument.

**See Also:** "Oracle Internet Directory Server Administration Tools" in Oracle Identity Management User Reference for syntax and usage notes for each of the command-line tools.

**Table D–3** Examples: Using the -E Argument with Command-Line LDAP Tools

<table>
<thead>
<tr>
<th>Tool</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>ldapbind</td>
<td>ldapbind -h my_host -p 389 -E \ZHS16GBK\ -D &quot;o=acme,c=us&quot; -w my_password</td>
</tr>
<tr>
<td>ldapsearch</td>
<td>ldapsearch -h my_host -p 389 -E \ZHS16GBK\ -D &quot;o=acme,c=us&quot; -w my_password</td>
</tr>
<tr>
<td>ldapadd</td>
<td>ldapadd -h my_host -p 389 -E \ZHS16GBK\ -D &quot;o=acme,c=us&quot; -w my_password</td>
</tr>
<tr>
<td>ldapaddmt</td>
<td>ldapaddmt -h my_host -p 389 -E \ZHS16GBK\ -D &quot;o=acme,c=us&quot; -w my_password</td>
</tr>
<tr>
<td>ldapmodify</td>
<td>ldapmodify -h my_host -p 389 -E \ZHS16GBK\ -D &quot;o=acme,c=us&quot; -w my_password</td>
</tr>
</tbody>
</table>
Setting NLS_LANG in the Client Environment

If the output required by the client is UTF-8, then you do not need to set the NLS_LANG environment variable. In this case, the character set component of the NLS_LANG environment variable defaults to AL32UTF8, and both the input path from client to server, and the output path from server to client, do not require any character set conversion.

If the output required by the client is not UTF-8, then you must set the NLS_LANG environment variable. This ensures that proper character set conversion can occur from the AL32UTF8 character set to the character set required by the client.

For example, if the NLS_LANG environment variable is set to the simplified Chinese character set, then the command-line tool displays output in that character set. Otherwise the output defaults to the AL32UTF8 character set.

Note: If you are using Microsoft Windows, then, to use the command-line tools after server startup, you must reset NLS_LANG in an MS-DOS window. Set it to the character set that matches the code page of your MS-DOS session. AL32UTF8 cannot be used. See the Oracle Database Installation Guide for Microsoft Windows (32-Bit) for more information on which character set to use for command-line tools in an MS-DOS session.

If you are using a pre-installed Oracle Database with Oracle Internet Directory, then you must also set the database character set to AL32UTF8.

See Also: Oracle Database Globalization Support Guide in the Oracle Database Documentation Library and Oracle Database Installation Guide for Microsoft Windows (32-Bit)

Be careful not to change the NLS_LANG parameter value in the registry.
Using Globalization Support with Bulk Tools

Oracle Internet Directory ensures that the reading and writing of text data from and to LDIF files are done in UTF-8 encoding as specified by the LDAP standard.

This section provides an example of the argument you use for each of the following bulk tools:

- Using Globalization Support with bulkload
- Using Globalization Support with ldifwrite
- Using Globalization Support with bulkdelete
- Using Globalization Support with bulkmodify

See Also: "Oracle Internet Directory Server Administration Tools" in Oracle Identity Management User Reference for a list of arguments for each bulk tool

Using Globalization Support with bulkload

Add to the command the argument `encode="character_set"` where the input LDIF file is encoded in "character_set".

For example:

```
bulkload connect="connect_string" encode=".ZHS16GBK" check="TRUE" \
generate="TRUE" file="my_ldif_file"
```

Using Globalization Support with ldifwrite

The ldifwrite utility always writes BASE64 encoded values for multibyte strings. The BASE64 encoding could be of the UTF-8 strings as they are stored in the directory server, or of native strings as specified by the `NLS_LANG` environment variable setting when running ldifwrite.

For example:

```
ldifwrite connect="connect_string" basedn="baseDN" file="output_file"
```

In this example, if the `NLS_LANG` environment variable is not set, or is set to `language_territory.AL32UTF8`, then the output LDIF file will contain BASE64-encoded UTF-8 strings for any multibyte characters.

To reload this LDIF file into the directory by using ldapaddmt, use the following syntax:

```
ldapaddmt -h my_host -p port_number -f output_file
```

In this case, the `-E` argument is not required because the decoded BASE64 strings are already UTF-8-encoded and can be readily sent to the server.

If the `NLS_LANG` environment variable is set to a character set other than `AL32UTF8`—for example, `".ZHS16GBK"`—then the output LDIF file will contain a BASE64 encoded value of simplified Chinese GBK strings.

To reload this LDIF file into the directory using ldapaddmt, use the following syntax:

```
ldapaddmt -h host -p port -E ".ZHS16GBK" -f my_input_file.LDIF
```

See Also:
"Oracle Internet Directory Server Administration Tools" in Oracle Identity Management User Reference for a list of arguments for each bulk tool
In this case the –E argument is required because the decoded BASE64 strings are simplified Chinese GBK, which need to be converted to UTF-8 strings before being sent to the server.

**Using Globalization Support with bulkdelete**

Add `encode="character_set"` to the command.

For example:

```
bulkdelete connect=connect_string encode=\.ZHS16GBK" \
    base=ou=manufacturing,o=acme,c=us"
```

In this case the value for the –base option could be in the ZHS16GBK native character set, that is, simplified Chinese.

**Using Globalization Support with bulkmodify**

Add `encode="character_set"` to the command the argument.

For example:

```
bulkmodify connect="my_service_name" encode=".ZHS16GBK" \
    basedn=ou=manufacturing,o=acme,c=us" replace=title" \
    value=Foreman filter=objectclass=*"
```

In this example, values for the basedn, value, and filter arguments can be specified using the simplified Chinese GBK character set.
Setting up Access Controls for Creation and Search Bases for Users and Groups

If you modify the User Search Base, the User Creation Base, the Group Search Base, or the Group Creation Base, then access controls for the new container need to be set up properly. This appendix contains these topics:

- Setting up Access Controls for the User Search Base and the User Creation Base
- Setting up Access Controls for the Group Search Base and the Group Creation Base

Setting up Access Controls for the User Search Base and the User Creation Base

To set up access controls for the User Search Base and the User Creation Base:

1. Create an LDIF (user_aci.ldif) file with the following entry:

```ldif
--- BEGIN LDIF file contents---
dn: %usersearch_or_createbase_dn%
changetype: modify
add: orclaci
 orclaci: access to entry by group="cn=oracledascreateuser,
cn=groups,cn=OracleContext,%subscriberdn%"
   added_object_constraint=(objectclass=orcluser*) (browse,add) by
group="cn=Common User Attributes, cn=Groups,
cn=OracleContext,%subscriberdn%" (browse) by
   group="cn=PKIAdmins, cn=groups, cn=OracleContext,%subscriberdn%" (browse)
orclaci: access to entry filter=(objectclass=inetorgperson) by
group="cn=oracledascreateuser, cn=groups,cn=OracleContext,%subscriberdn%"
   added_object_constraint=(objectclass=orcluser*) (browse,add) by
group="cn=oracledasdeleteuser, cn=groups,cn=OracleContext,%subscriberdn%"
   (browse,delete) by group="cn=oracledasedituser,
cn=groups,cn=OracleContext,%subscriberdn%" (browse)
   group="cn=UserProxyPrivilege, cn=Groups,cn=OracleContext,%subscriberdn%"
   (browse, proxy) by dn="orclApplicationCommonName=DASApp, cn=DAS,
cn=Products,cn=oraclecontext" (browse,proxy) by self (browse, nodelete, noadd)
   by
group="cn=Common User Attributes, cn=Groups,cn=OracleContext,%subscriberdn%"
   (browse) by * (browse, noadd, nodelete)
orclaci: access to attr=() filter=(objectclass=inetorgperson) by
group="cn=oracledasedituser, cn=groups,cn=OracleContext,
%subscriberdn%" (read,search,write,compare) by self (read,search,write,selfwrite,compare) by *
   (read, noWrite, noCompare)
--- END LDIF file contents---
```

Setting up Access Controls for the Group Search Base and the Group Creation Base

To set up access controls for the Group Search Base and the Group Creation Base:

2. Create an LDIF (group_aci.ldif) file with the following entry:

```ldif
--- BEGIN LDIF file contents---
dn: %groupsearch_or_createbase_dn%
changetype: modify
add: orclaci
 orclaci: access to entry by group="cn=oracledascreategroup,
cn=groups,cn=OracleContext,%subscriberdn%"
   added_object_constraint=(objectclass=orclgroup*) (browse,add) by
group="cn=Common Group Attributes, cn=Groups,
cn=OracleContext,%subscriberdn%" (browse) by
   group="cn=PKIAdmins, cn=groups, cn=OracleContext,%subscriberdn%" (browse)
orclaci: access to entry filter=(objectclass=inetorgperson) by
group="cn=oracledascreategroup, cn=groups,cn=OracleContext,%subscriberdn%"
   added_object_constraint=(objectclass=orcluser*) (browse,add) by
group="cn=oracledasdeletegroup, cn=groups,cn=OracleContext,%subscriberdn%"
   (browse,delete) by group="cn=oracledaseditgroup,
cn=groups,cn=OracleContext,%subscriberdn%" (browse)
   group="cn=UserProxyPrivilege, cn=Groups,cn=OracleContext,%subscriberdn%"
   (browse, proxy) by dn="orclApplicationCommonName=DASApp, cn=DAS,
cn=Products,cn=oraclecontext" (browse,proxy) by self (browse, nodelete, noadd)
   by
group="cn=Common Group Attributes, cn=Groups,cn=OracleContext,%subscriberdn%"
   (browse) by * (browse, noadd, nodelete)
orclaci: access to attr=() filter=(objectclass=inetorgperson) by
group="cn=oracledaseditgroup, cn=groups,cn=OracleContext,
%subscriberdn%" (read,search,write,compare) by self (read,search,write,selfwrite,compare) by *
   (read, noWrite, noCompare)
--- END LDIF file contents---
```
Setting up Access Controls for the User Search Base and the User Creation Base

orclaci: access to attr=(userPassword)
  filter=(objectclass/inetorgperson) by
group="cn=OracleUserSecurityAdmins,cn=Groups,cn=OracleContext,%subscriberdn%"
  (read, search, write, compare) by group="cn=oracledasedituser,cn=groups,cn=OracleContext,%subscriberdn%"
  (read, search, write, compare) by self
  (read, search, write, selfwrite, compare) by group="cn=authenticationServices,cn=Groups,cn=OracleContext,%subscriberdn%"
  (compare) by * (none)
orclaci: access to attr=(authpassword, orclpassword, orclpasswordverifier, orclpassword)
  filter=(objectclass=inetorgperson) by
group="cn=oracledasedituser,cn=groups,cn=OracleContext,%subscriberdn%"
  (read, search, write, compare) by group="cn=verifierServices,cn=Groups,cn=OracleContext,%subscriberdn%"
  (search, read, compare) by self
  (read, search, write, compare) by * (none)
orclaci: access to attr=(orclpwdaccountunlock)
  filter=(objectclass=inetorgperson) by
group="cn=oracledasedituser,cn=groups,cn=OracleContext,%subscriberdn%"
  (write) by * (none)
orclaci: access to attr=(usercertificate, usersmimecertificate)
  filter=(objectclass=inetorgperson) by
group="cn=PKIAdmins,cn=Groups,cn=OracleContext,%subscriberdn%"
  (read, search, write, compare) by self
  (read, search, compare) by *
  (read, search, compare)
orclaci: access to attr=(mail)
  filter=(objectclass=inetorgperson) by
group="cn=EmailAdminsGroup,cn=EmailServerContainer,cn=Products,cn=OracleContext"
  (write) by group="cn=oracledasedituser,cn=groups,cn=OracleContext,%subscriberdn%"
  (read, search, write, compare) by group="cn=Common User Attributes,cn=Groups,cn=OracleContext,%subscriberdn%"
  (read, search, compare)
  (read, search, write, compare) by * (read, nowrite, nocompare)
orclaci: access to attr=(orclguid, orclisenabled, modifytimestamp, mail)
  filter=(objectclass=inetorgperson) by
group="cn=Common User Attributes,cn=Groups,cn=OracleContext,%subscriberdn%"
  (read, search, compare)
  (read, nowrite, nocompare)
orclaci: access to attr=(orclpasswordhintanswer)
  filter=(objectclass=inetorgperson) by
group="cn=Common User Attributes,cn=Groups,cn=OracleContext,%subscriberdn%"
  (read, search, compare)
  (read, nowrite, nocompare)
orclaci: access to attr=(orclpasswordhint)
  filter=(objectclass=inetorgperson) by
group="cn=Common User Attributes,cn=Groups,cn=OracleContext,%subscriberdn%"
  (read, search, compare)
  (read, nowrite, nocompare)
orclaci: access to attr=(displayName, preferredlanguage, orcltimezone, orcldateofbirth, orclgender, orclwirelessaccountnumber, cn, uid, homephone, telephonenumber)
  filter=(objectclass=inetorgperson) by
group="cn=Common User Attributes,cn=Groups,cn=OracleContext,%subscriberdn%"
  (read, search, compare)
  (read, nowrite, nocompare)
orclaci: access to attr=(orclentrylevelaci)
  filter=(orcluser*)
  (browse, add) by * (browse)
---END LDIF file contents------

2. Replace %subscriberdn% with the dn of the subscriber and %usersearch_or_createbase_dn% with the new value of the container DN where the new user search/create base points to.
3. Run the ldapmodify command as follows:

```
ldapmodify -p oidport -h oidhost -D cn=orcladmin -w Instance Password -v \
-f user_aci.ldif
```

**Setting up Access Controls for the Group Search Base and the Group Creation Base**

To set up access controls for the Group Search Base and the Group Creation Base:

1. Create an ldif (group_aci.ldif) file with the following entry:

```
--- BEGIN LDIF file contents---
dn: %groupsearch_or_createbase_dn%
changetype: modify
add: orclaci
orclaci: access to entry by group="cn=IASAdmins,
cn=groups,cn=OracleContext,%subscriberdn%"
    added_object_constraint=(objectclass=orclcontainer) (browse,add)
orclaci: access to entry by group="cn=oracledascreategroup,
cn=groups,cn=OracleContext,%subscriberdn%"
    added_object_constraint=(objectclass=orclgroup*) (browse,add) by
group="cn=Common Group Attributes, cn=Groups,cn=OracleContext,%subscriberdn%" (browse) by * (none)
orclaci: access to entry
    filter=(&(objectclass=orclgroup)(!(orclisvisible=false))) by
groupattr=(owner) (read,search,write,compare) by * (read, search, compare)
orclaci: access to attr=(*)
    filter=(objectclass=orclgroup)(orclisvisible=false)) by
groupattr=(owner) (read,search,write,compare) by * (read, search, compare) by 
group="cn=Common Group Attributes, cn=Groups,cn=OracleContext,%subscriberdn%" (read, search, compare) by groupattr=(owner) (read, search, compare)
add: orclentrylevelaci
orclentrylevelaci: access to entry by group="cn=oracledascreategroup,
cn=groups,cn=OracleContext,%subscriberdn%"
    added_object_constraint=(objectclass=orclgroup) (browse, add) by
    group="cn=IASAdmins, cn=groups,cn=OracleContext,%subscriberdn%"
    added_object_constraint=(objectclass=orclcontainer) (browse,add) by * (browse)
---END LDIF file contents------
```
2. Replace \%subscriberdn\% with the DN of the subscriber and \%groupsearch_or_createbase_dn\% with the new value of the container DN where the new group search base or group create base points to.

3. Run the ldapmodify command as follows:

```bash
ldapmodify -p oidport -h oidhost -D cn=orcladmin -w instance password \
    -v -f group_aci.ldif
```
This appendix describes how the multimaster replication process adds, deletes, and modifies entries, and how it modifies DNs and RDNs. It contains these topics:

- How the Multimaster Replication Process Adds a New Entry to a Consumer
- How the Multimaster Replication Process Deletes an Entry
- How the Multimaster Replication Process Modifies an Entry
- How the Multimaster Replication Process Modifies a Relative Distinguished Name
- How the Multimaster Replication Process Modifies a Distinguished Name

How the Multimaster Replication Process Adds a New Entry to a Consumer

When the directory replication server adds a new entry to a consumer, it follows this change application process:

1. The directory replication server looks in the consumer for the DN of the parent of the target entry. Specifically, it does this by looking for a global unique identifier (GUID) assigned to the DN of the parent.

2. If the parent entry exists, then the directory replication server composes a DN for the new entry and places the new entry under its parent in the consumer. It then places the change entry in the purge queue.

   If the change entry is not successfully applied on the first try, then:
   
The directory replication server places the new change entry in the retry queue, sets the number of retries to the configured maximum, and repeats the change application process.

   If the change entry is not successfully applied on all but the last retry, then:
   
The directory replication server keeps the change entry in the retry queue, decrements the number of retries, and repeats the change application process.

   If the change entry is not successfully applied on the last retry, then:
   
The directory replication server checks to see if the new entry is a duplicate of an existing entry.

   If the change entry is a duplicate entry, then:
   
The directory replication server applies the following conflict resolution rules:
   
   - The entry with the older creation time stamp is used.
How the Multimaster Replication Process Deletes an Entry

- If both entries have the same creation time stamp, then the entry with the smaller GUID is used.

If the change entry is used, then the target entry is removed, the change is applied, and the change entry is placed in the purge queue.

If the target entry is used, then the change entry is placed in the purge queue.

**If the change entry is not a duplicate entry, then:**

The directory replication server places the change entry in the human intervention queue, and repeats the change application process at the interval you specified in the `orclHIQSchedule` parameter.

**If the change entry is not successfully applied after it has been placed in the human intervention queue:**

The directory replication server keeps the change in this queue, and repeats the change application process at specified intervals while awaiting action by the administrator. The administrator can use the Oracle Internet Directory Comparison and Reconciliation Tool and the human intervention queue manipulation tool to resolve the conflict.

---

How the Multimaster Replication Process Modifies an Entry

When the directory replication server modifies an entry in a consumer, it follows this change application process:

1. The directory replication server looks in the consumer for an entry with a GUID matching the one in the change entry.

2. If the matching entry exists in the consumer, then the directory replication server deletes it. It then places the change entry in the purge queue.

**If the change entry is not successfully applied on the first try, then:**

The directory replication server places the change entry in the retry queue, sets the number of retries to the configured maximum, and repeats the change application process.

**If the change entry is not successfully applied on all but the last retry, then:**

The directory replication server keeps the change entry in the retry queue, decrements the number of retries, and repeats the change application process.

**If the change entry is not successfully applied on the last retry, then:**

The directory replication server places the change entry in the human intervention queue and repeats the change application process at specified intervals.

**If the change entry is not successfully applied after it has been placed in the human intervention queue:**

The directory replication server keeps the change entry in this queue, and repeats the change application process at specified intervals while awaiting action by the administrator. The administrator can use the Oracle Internet Directory Comparison and Reconciliation Tool and the human intervention queue manipulation tool to resolve the conflict.

---

How the Multimaster Replication Process Modifies an Entry

When the directory replication server modifies an entry in a consumer, it follows this change application process:
1. The directory replication server looks in the consumer for an entry with a GUID matching the one in the change entry.

2. If the matching entry exists in the consumer, then the directory replication server compares each attribute in the change entry with each attribute in the target entry.

3. The directory replication server then applies the following conflict resolution rules:
   a. The attribute with the most recent modify time is used.
   b. The attribute with the most recent version of the attribute is used—for example, version 1, 2, or 3.
   c. The modified attribute on the host whose name is closest to the beginning of the alphabet is used.

4. The directory replication server applies the filtered modification, and places the change entry in the purge queue.

If the change entry is not successfully applied on the first try, then:

The directory replication server places the change entry in the retry queue, sets the number of retries to the configured maximum, and repeats the change application process.

If the change entry is not successfully applied on all but the last retry, then:

The directory replication server keeps the change entry in the retry queue, decrements the number of retries, and repeats the change application process.

If the change entry is not successfully applied by the last retry, then:

The directory replication server places the change entry in the human intervention queue and repeats the change application process at specified intervals.

If the change entry is not successfully applied after it has been placed in the human intervention queue:

The directory replication server keeps the change entry in this queue, and repeats the change application process at specified intervals while awaiting action by the administrator. You can use the Oracle Internet Directory Comparison and Reconciliation Tool and the Human Intervention Queue Manipulation Tool to resolve the conflict.

---

**How the Multimaster Replication Process Modifies a Relative Distinguished Name**

When the directory replication server modifies the RDN of an entry in a consumer, it follows this change application process:

1. The directory replication server looks in the consumer for the DN with a GUID that matches the GUID in the change entry.

2. If the matching entry exists in the consumer, then the directory replication server modifies the RDN of that entry and places the change entry in the purge queue.

If the change entry is not successfully applied on the first try, then:

The directory replication server places the change entry in the retry queue, sets the number of retries to the configured maximum, and repeats the change application process.

If the change entry is not successfully applied on all but the last retry, then:
The directory replication server keeps the change entry in the retry queue, decrements the number of retries, and repeats the change application process.

If the change entry is not successfully applied on the last retry, then:

The directory replication server places the change entry in the human intervention queue and checks to see if it is a duplicate of the target entry.

If the change entry is a duplicate entry, then:

The directory replication server applies the following conflict resolution rules:

- The entry with the older creation time stamp is used.
- If both entries have the same creation time stamp, then the entry with the smaller GUID is used.

If the change entry is used, then the target entry is removed, the change entry is applied, and then placed in the purge queue.

If the target entry is used, then the change entry is placed in the purge queue.

If the change entry is not a duplicate entry, then:

The directory replication server places the change entry in the human intervention queue, and repeats the change application process at specified intervals.

If the change entry is not successfully applied after it has been placed in the human intervention queue:

The directory replication server keeps the change entry in this queue, and repeats the change application process at specified intervals while awaiting action by the administrator. The administrator can use the Oracle Internet Directory Comparison and Reconciliation Tool and the Human Intervention Queue Manipulation Tool to resolve the conflict.

---

**How the Multimaster Replication Process Modifies a Distinguished Name**

When the directory replication server modifies the DN of an entry in a consumer, it follows this change application process:

1. The directory replication server looks in the consumer for the DN with a GUID that matches the GUID in the change entry.
   
   The directory replication server also looks in the consumer for the parent DN with a GUID that matches the GUID of the new parent specified in the change entry.

2. If both the DN and the parent DN of the target entry exist in the consumer, then the directory replication server modifies the DN of that entry and places the change entry in the purge queue.

If the change entry is not successfully applied on the first try, then:

The directory replication server places the change entry in the retry queue, sets the number of retries to the configured maximum, and repeats the change application process.

If the change entry is not successfully applied on all but the last retry, then:

The directory replication server keeps the change entry in the retry queue, decrements the number of retries, and repeats the change application process.

If the change entry is not successfully applied by the last retry, then:
The directory replication server places the change entry in the human intervention queue and checks to see if it is a duplicate of the target entry.

**If the change entry is a duplicate entry, then:**

The directory replication server applies the following conflict resolution rules:

- The entry with the older creation time stamp is used.
- If both entries have the same creation time stamp, then the entry with the smaller GUID is used.

If the change entry is used, then the target entry is removed, the change entry is applied, and then placed in the purge queue.

If the target entry is used, then the change entry is placed in the purge queue.

**If the change entry is not a duplicate entry, then:**

The directory replication server places the change entry in the human intervention queue, and repeats the change application process at specified intervals.

**If the change entry is not successfully applied after it has been placed in the human intervention queue:**

The directory replication server keeps the change entry in this queue, and repeats the change application process at specified intervals while awaiting action by the administrator. The administrator can use the Oracle Internet Directory Comparison and Reconciliation Tool and the Human Intervention Queue Manipulation Tool to resolve the conflict.
Searching the Directory for User Certificates

Starting with 10g (10.1.4.0.1), you can perform a command-line search of the binary attribute `usercertificate`.

Prior to 10g Release 2 (10.1.2.0.2), the only way to identify a user from the certificate was through the DN specified in the certificate. This is known as certificate matching. Starting with 10g Release 2 (10.1.2.0.2), Oracle Internet Directory supports certificate mapping, in addition to certificate matching. Certificate matching requires that a user certificate be provisioned in the directory. Certificate mapping does not require provisioning of a user certificate.

This chapter includes the following topics:

- **Certificate Mapping**
- **Search Types**

**Certificate Mapping**

Certificate mapping allows a customer to define rules for mapping the certificate to the user's DN. A certificate mapping rule is a set of rules for parsing the certificate and for querying the directory for the user's identity. Only custom extensions of certificates can be used in mapping rules.

The following examples show how to add, delete, and modify a certificate mapping rule.

**Adding a Certificate Mapping Rule**

Add a mapping rule using `ldapmodify`, as follows:

```
ldapmodify -h hostName -p port_number -f certMapRuleAdd.ldif
```

The file `certMapRuleAdd.ldif` should look something like this:

```
dn: cn=maprule1,cn=SASL-EXTERNAL,cn=Identity Mapping Configurations,cn=Server Configurations
cn: maprule1
objectclass: orclidmapping
objectclass: orclcertidmapping
orclSearchScope: subtree
orclSearchFilter: (cn=$(2.16.750.5.14.2.81.2.5.1))
orclcertExtensionOID: 2.16.750.5.14.2.81.2.5
orclcertExtensionAttribute: 2.16.750.5.14.2.81.2.5.1
```
Deleting a Certificate Mapping Rule
Delete a mapping rule using ldapdelete, as follows:

```bash
ldapdelete hostName -p port_number "cn=maprule1,cn=SASL-EXTERNAL,cn=Identity Mapping Configurations,cn=Server Configurations"
```

Modifying a Certificate Mapping Rule
Modify a mapping rule using ldapmodify, as follows:

```bash
ldapmodify -h hostName -p port_number -f certMapRuleMod.ldif
```

The file certMapRuleMod.ldif should look something like this:

```bash
dn: cn=maprule1,cn=SASL-EXTERNAL,cn=Identity Mapping Configurations,cn=Server Configurations
changetype:modify
replace: <attrName>
<attrName>: <attrValue>
```

Search Types

You can use two kinds of ldapsearch filters:

- A filter of the form "usercertificate=certificate_serial_number$certificate_issuer_DN". A combination of the certificate serial number and the certificate issuer's DN is used to locate the certificate. This combination is called the certificate match value.

- A filter of the form"usercertificate;binary=base_64_encoded_value_of_certificate". Using this filter, one of six types of searches is possible, depending upon two things:
  - The value of the DSA configuration set attribute (DN: "cn=dsacfg, cn=configsets, cn=oracle internet directory"), orclpkimatchingrule.
  - The presence or absence of the LDAP control 2.16.840.1.113894.1.8.23

The six types of searches possible with a filter of the form "usercertificate;binary=base_64_encoded_value_of_certificate" are:

<table>
<thead>
<tr>
<th>Presence of LDAP control</th>
<th>Value of orclpkimatchingrule</th>
<th>Search Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absent</td>
<td>Not used</td>
<td>The hashed value of the client certificate is used to locate usercertificate.</td>
</tr>
<tr>
<td>Present</td>
<td>0</td>
<td>An exact-match search is performed. The subject DN of the client certificate is the search base. This DN is compared with the user DN in the directory. The search scope is Base. The filter is &quot;objectclass=*&quot;.</td>
</tr>
<tr>
<td>Present</td>
<td>1</td>
<td>The hashed value of the client certificate is used to locate usercertificate.</td>
</tr>
<tr>
<td>Present</td>
<td>2 (Default)</td>
<td>The hashed value of the client certificate is used to locate usercertificate. If this search yields nothing, An exact-match search is performed.</td>
</tr>
</tbody>
</table>
Use the `ldapmodify` tool to set `orclpkimatchingrule` to the desired value.

<table>
<thead>
<tr>
<th>Presence of LDAP control</th>
<th>Value of orclpkimatchingrule</th>
<th>Search Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>Present</td>
<td>3</td>
<td>The mapping rule is used.</td>
</tr>
<tr>
<td>Present</td>
<td>4</td>
<td>First, the mapping rule is used. If that search yields nothing, then the search proceeds as if the value were 2.</td>
</tr>
</tbody>
</table>

**Notes:**

- The `usercertificate` attribute cannot be searched using a substring filter.
- In an exact-match search, the search filter can contain only one attribute value assertion.
- Only one-level and subtree searches are supported.
- The `catalog` tool does not support catalogs for user certificates—namely `ct_orclcertificatehash` and `ct_orclcertificatematch`.
- The introduction in 10g (10.1.4.0.1) of certificate hash values requires that certificates be upgraded from earlier releases. See the `upgradecert.pl` command-line tool reference in *Oracle Identity Management User Reference*.

**See Also:** "Direct Authentication" on page 16-4
This chapter describes the replica states for LDAP-based replication. LDAP replica states have no effect upon Advanced Replication.

When LDAP based replication is configured, and the replication server is started, the server reads the replica state `orclreplicastate` from the local replica, "orclreplicaid=local_Replcia_ID, cn=replication configuration". The replication server behaves differently, based upon the local replica state, as shown in Table H–1. The replication server reads the local replica state from the local (consumer) node.

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
<th>Replication Server Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Bootstrap</td>
<td>Starts replication bootstrap processing to synchronize the consumer directory from the supplier, based on the replication naming context configuration. Updates the replica state to correspond with bootstrap progress.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Sets the replica state to 3 (Bootstrap in progress) immediately when it starts to bootstrap.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Sets the replica state to 4 (Bootstrap in progress, <code>cn=oraclecontext</code> bootstrap has completed) after it completes bootstrapping of &quot;cn=oraclecontext&quot; successfully.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Sets the replica state to 5 (Bootstrap Error occurred) after bootstrap is completed but failure(s) detected during the bootstrap. Then it waits until the replica state is re-set. Note: Human intervention is required; See &quot;Troubleshooting Oracle Internet Directory Replication&quot; on page L-14 for details.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Sets the replica state to 1 (On-line) after bootstrap has completed successfully. Then replication will automatically start to perform normal replication processing.</td>
</tr>
<tr>
<td>1</td>
<td>On line</td>
<td>Starts normal replication processing to replicate changes from the supplier to the consumer.</td>
</tr>
<tr>
<td>2</td>
<td>Off line</td>
<td>Logs an error message in oidrepld.log similar to this:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2004/09/24:17:41:44 * Replica(dlsun1418_replica2) is in OFFLINE mode, Please update the replica state and restart OIDREPLD...</td>
</tr>
<tr>
<td></td>
<td></td>
<td>The administrator must set the replica state properly and restart the replication server.</td>
</tr>
<tr>
<td>3</td>
<td>Bootstrap in progress</td>
<td>Sets the replica state back to 0 (Bootstrap), then starts to bootstrap again as if the replica state were 0.</td>
</tr>
</tbody>
</table>
Oracle Internet Directory replication server logs the bootstrapping process in the $ORACLE_HOME/ldap/log/oidrepld00.log.

If bootstrap completes successfully, the log looks similar to the following example, and the replication server will automatically start to perform normal replication processing.

```
2004/10/06:17:13:25 * Starting OIDREPLD against isunnad03:5555...
2004/10/06:17:13:26 * Starting scheduler...
2004/10/06:17:13:27 * Start to BootStrap from supplier=isunnad03_purify to consumer=isunnad03_purify3
2004/10/06:17:13:28 * gslrbssSyncDIT:Replicating namingcontext=cn=oraclecontext ... ...
2004/10/06:17:14:21 * gslrbssSyncDIT:Sync done successfully for namingctx: cn=oraclecontext, 222 entries matched
2004/10/06:17:14:21 * gslrbssSyncDIT:Replicating namingcontext=c=india ... ...
2004/10/06:17:14:21 * gslrbssSyncDIT:Sync done successfully for namingctx: c=india, 0 entries matched
2004/10/06:17:14:21 * gslrbssSyncDIT:Replicating namingcontext=c=uk ... ...
2004/10/06:17:19:57 * gslrbssSyncDIT:Sync done successfully for namingctx: c=uk, 1087 entries matched
2004/10/06:17:19:57 * gslrbssSyncDIT:Replicating namingcontext=cn=oracleschemaversion ... ...
2004/10/06:17:19:58 * gslrbssSyncDIT:Sync done successfully for namingctx: cn=oracleschemaversion, 10 entries matched
2004/10/06:17:20:01 * gslrbsbBootStrap: BOOTSTRAP DONE SUCCESSFULL
```

If failures are detected, the log looks similar to the following example:

```
2004/09/14:12:57:23 * Starting OIDREPLD against disun1418:4444...
2004/09/14:12:57:25 * Starting scheduler...
2004/09/14:12:57:26 * Start to BootStrap from supplier=disun1418_replica to consumer=disun1418_replica2
2004/09/14:12:57:27 * gslrbssSyncDIT:Replicating namingcontext=cn=oraclecontext ... ...
```

Table H–1 (Cont.) LDAP Replica States

<table>
<thead>
<tr>
<th>Value</th>
<th>Meaning</th>
<th>Replication Server Behavior</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>Bootstrap in progress, cn=oraclecontext bootstrap has completed.</td>
<td>Sets the replica state back to 0 (Bootstrap), then starts to bootstrap again as if the replica state were 0.</td>
</tr>
<tr>
<td>5</td>
<td>Bootstrap completed; failure detected for one or more naming contexts.</td>
<td>Logs an error message in oidrepld.log similar to this: 2004/09/24:17:13:30 * Replication BOOTSTRAP_ERROR mode detected for replica(dlsun1418_replica2) Then it waits until the replica state is reset properly.</td>
</tr>
<tr>
<td>6</td>
<td>Database copy-Based addnode; used only in Oracle Database Advanced Replication-based replication</td>
<td>This mode indicates that the replica is a database copy-based addnode for Oracle Database Advanced Replication-based replication. This replica state is not to be used in LDAP-based replication.</td>
</tr>
</tbody>
</table>

Oracle Internet Directory replication server logs the bootstrapping process in the Oracle Internet Directory replication server log, $ORACLE_HOME/ldap/log/oidrepld00.log.
2004/09/14:12:58:21 * gslrbssSyncDIT: Replicating namingcontext=cn=quan zhou ......
2004/09/14:12:58:23 * gslrbssSyncDIT: Sync failed for namingctx: cn=quan zhou, only 1 entries retrieved
2004/09/14:12:58:23 * gslrbssSyncDIT: Replicating namingcontext=cn=oracleschemaversion ......
2004/09/14:12:58:25 * gslrbssSyncDIT: Sync done successfully for namingctx: cn=oracleschemaversion, 10 entries matched
2004/09/14:12:58:51 * gslrbsbBootStrap: Failure occurred when bootstrapping 1 out of 3 namingcontext(s) from the supplier

**Tip:** You have two options for troubleshooting bootstrap failure.

- **Option 1:** Identify the cause of the bootstrap failure and fix the cause, then restart bootstrapping by setting the consumer replica’s `orclreplicastate` to **Bootstrap mode**.

- **Option 2:** Identify the naming contexts that failed to be bootstrapped and use `oidcmprec` to reconcile them. Then resume replication by setting the consumer replica’s `orclreplicastate` to **Online mode**.

**Note:** `Oidrepld` is now in **Bootstrap_error** mode, so you will need to reset the consumer replica’s replica state (`orclreplicastate`).
Addition of a Directory Node by Using the Database Copy Procedure

This appendix explains how to add a new node to an existing replicating system by using the database copy procedure, also known as cold backup. This procedure works only for Oracle Internet Directory. Do not employ this procedure if other Oracle Identity Management components, such as Oracle Application Server Certificate Authority or Oracle Application Server Single Sign-On, are installed. You can use the database copy procedure, to create a new Directory Replication Group if you have a standalone Oracle Internet Directory node.

This procedure is applicable for an Oracle Database Advanced Replication-based replica and for a full LDAP-based replica. The steps for Advanced replication are slightly different from the steps for LDAP replication. When adding the node, be sure to use the procedure described in Addition of an Oracle Advanced Database Replication-Based Directory Node for an Oracle Database Advanced Replication-based replica or the procedure described in Addition of an LDAP Replication-Based Directory Node for a full LDAP-based replica.

This chapter contains these topics:

- Definitions
- Prerequisites
- Sponsor Directory Site Environment
- New Directory Site Environment
- Preliminary Tasks To Be Performed on the New Node
- Addition of an Oracle Advanced Database Replication-Based Directory Node
- Addition of an LDAP Replication-Based Directory Node

Definitions

The sponsor site refers to the site or host or node where Oracle Internet Directory and its repository, the Oracle database, are installed. The sponsor site is also referred to as sponsor node.

The new site refers to the site or host or node to which you are copying the Oracle Internet Directory repository. The new site is also referred to as the new node.
Prerequisites

Your computing environment must meet the following prerequisites before you start this procedure:

1. The operating system, version, and patch level of the new directory site must be the same as that of the sponsor directory site. This procedure might not work if the patch level of the operating system differs.

2. Oracle Corporation strongly recommends that you back up the sponsor directory's repository before you employ this procedure.

3. Because this procedure involves copying Oracle data files, performance depends on the underlying network. If the underlying network is slow, it might be more efficient to use the method described in Chapter 30, "Oracle Internet Directory Replication Installation and Configuration" to set up a replication group. Alternatively, you could physically transfer compressed Oracle data files on removable media. Consult your local system or network administrator for information about the network.

4. Only a person familiar with the Oracle database should perform this procedure.

Sponsor Directory Site Environment

In the example shown throughout this chapter, the sponsor directory site's environment is as follows:

Hostname = rst-sun
Domain name = acme.com
ORACLE_BASE = /private/oracle/app/oracle
ORACLE_HOME = /private/oracle/app/oracle/product/OraHome_1
ORACLE_SID = LDAP
LD_LIBRARY_PATH = $ORACLE_HOME/lib
NLS_LANG = AMERICAN_AMERICA.AL32UTF8
datafile location = /private/oracle/oradata/LDAP
Dump destination = /private/oracle/app/oracle/admin/LDAP/pfile,
                   /private/oracle/app/oracle/admin/LDAP/bdump,
                   /private/oracle/app/oracle/admin/LDAP/cdump,
                   /private/oracle/app/oracle/admin/LDAP/udump,
                   /private/oracle/app/oracle/admin/LDAP/create

New Directory Site Environment

In the example shown throughout this chapter, the new directory site's environment is as follows:

Hostname = dsm-sun
Domain name = acme.com
ORACLE_BASE = /private1/oracle/app/oracle
ORACLE_HOME = /private1/oracle/app/oracle/product/OraHome_1
ORACLE_SID = NLDAP
LD_LIBRARY_PATH = $ORACLE_HOME/lib
NLS_LANG = AMERICAN_AMERICA.UTF8
datafile location = /private1/oracle/oradata/NLDAP
Dump destination = /private1/oracle/app/oracle/admin/NLDAP/pfile,
                   /private1/oracle/app/oracle/admin/NLDAP/bdump,
                   /private1/oracle/app/oracle/admin/NLDAP/cdump,
                   /private1/oracle/app/oracle/admin/NLDAP/udump,
                   /private1/oracle/app/oracle/admin/NLDAP/create
Preliminary Tasks To Be Performed on the New Node

Complete the following steps on the new node:

1. Log in to the new node, dsm-sun.

2. Install Identity Management and Metadata Repository using Oracle Universal Installer. You need to install only Oracle Internet Directory, so select only Oracle Internet Directory in the Select Configuration Options screen of the installer. During the installation process, set ORACLE_SID to NLDAP and global name to NLDAP.ACME.COM.

   See Also: "Installing OracleAS Infrastructure" in Oracle Application Server Installation Guide.

3. Issue the following command to copy all install-specific entries to sponsor directory:

   remtool -backupmetadata \
   -replica "new_node_host:new_node_port/new_node_repldn_pwd" \
   -master "sponsor_host:master_port/sponsor_repl_dn_pwd"

   where sponsor_host, sponsor_port, and sponsor_repl_dn_pwd are the hostname, port number, and replication DN password for the sponsor node, respectively.

   Note: If Oracle Delegated Administration Services is not configured, you might see an error message similar to this when you run remtool with the -backupmetadata option:

   Failed to add "orclApplicationCommonName=ias.acme.com, \
   cn=IAS Instances, cn=IAS, cn=Products, cn=OracleContext" \
   as "uniquemember" to entry "cn=Associated Mid-tiers, \
   orclapplicationcommonname=DASApp, cn=DAS, cn=products, \
   cn=OracleContext at replica ldap://myhost:389"

   Please ignore this error message.

   See Also: "If an Existing Master is Used as a Remote Master Site" on page 30-8 in Chapter 30, "Oracle Internet Directory Replication Installation and Configuration"

Addition of an Oracle Advanced Database Replication-Based Directory Node

This section describes the addition of an Advanced Replication-based directory node. It contains the following sections:

- Tasks To Be Performed on the Sponsor Advanced Replication Node
- Tasks To Be Performed on the New Advanced Replication Node
- Verification of an Advanced Replication-Based Replica Node

Tasks To Be Performed on the Sponsor Advanced Replication Node

Complete the following steps on the sponsor node.

1. At the command line prompt execute SQL*Plus.
Addition of an Oracle Advanced Database Replication-Based Directory Node

$ sqlplus /nolog
SQL> connect /as sysdba
SQL> ALTER DATABASE BACKUP CONTROLFILE TO TRACE RESETLOGS;

This command creates a trace file under the user dump destination directory. In this example, that directory is /private/oracle/app/oracle/admin/LDAP/udump.

The file will be created in the following format:
$ORACLE_SID_ora_processid.trc

For example:
ldap_ora_4765.trc

2. Shut down the LDAP and replication servers and OID Monitor processes. Make sure the LDAP and replication servers are stopped before stopping the OID Monitor process.

   $ oidctl connect=connect_string server=oidrepld instance=instance_number stop
   $ oidctl connect=connect_string server=oidldapd instance=instance_number stop
   $ oidmon connect=connect_string stop

You can also stop OIDMON and the LDAP server process using opmn. Ensure that the replication server, directory server and OIDMON are shut down before you proceed to next step.

In these commands, connect_string is the net service name in the node's tnsnames.ora file.

3. On the other nodes of the directory replication group (DRG), shut down the LDAP replication server only.

   $ oidctl connect=connect_string server=oidrepld instance=instance_number stop

Repeat this procedure on all nodes except the sponsor node. Specify appropriate net service names for the corresponding nodes.

4. Perform this step only if adding a node to an existing DRG. Quiesce Oracle Database Advanced Replication by running the following commands at the master definition site (MDS):

   cd $ORACLE_HOME/ldap/admin

At the command line prompt execute SQL*Plus.

   $ sqlplus /nolog
   SQL> connect repadmin/repadmin_password;
   SQL> @oidrsusp.sql

   At this point, other nodes are available for LDAP edits, but replication will not occur.

   5. Shut down the database and Oracle Net Services listener on the sponsor node only:

   $ lsnrctl [listener_name] stop
   $ sqlplus /nolog

---

Note: Perform this procedure only on the master definition site.

---

At this point, other nodes are available for LDAP edits, but replication will not occur.

---

I-4  Oracle Internet Directory Administrator's Guide
Addition of an Oracle Advanced Database Replication-Based Directory Node

Addition of a Directory Node by Using the Database Copy Procedure

I-5

SQL> connect /as sysdba
SQL> shutdown normal
SQL> exit

By default, the listener name is LISTENER.

6. Copy the trace file created under Step 1 to a new file, newdb.sql, under the same directory.

$ cd $ORACLE_BASE/admin/LDAP/udump
$ cp ldap_ora_4765.trc newdb.sql

7. Edit newdb.sql, using any text editor, and delete all the lines except the STARTUP NOMOUNT and CREATE CONTROLFILE statements. After editing, newdb.sql should look like this:

STARTUP NOMOUNT
CREATE CONTROLFILE REUSE SET DATABASE "LDAP" RESETLOGS NOARCHIVELOG
MAXLOGFILES 16
MAXLOGMEMBERS 3
MAXDATAFILES 100
MAXINSTANCES 8
MAXLOGHISTORY 454
LOGFILE
  GROUP 1 '/private/oracle/oradata/LDAP/redo01.log' SIZE 10M,
  GROUP 2 '/private/oracle/oradata/LDAP/redo02.log' SIZE 10M,
  GROUP 3 '/private/oracle/oradata/LDAP/redo03.log' SIZE 10M
-- STANDBY LOGFILE
DATAFILE
  '/private/oracle/oradata/LDAP/system01.dbf',
  '/private/oracle/oradata/LDAP/sysaux01.dbf',
  '/private/oracle/oradata/LDAP/users01.dbf',
  '/private/oracle/oradata/LDAP/dcm.dbf',
  '/private/oracle/oradata/LDAP/portal.dbf',
  '/private/oracle/oradata/LDAP/ptldoc.dbf',
  '/private/oracle/oradata/LDAP/ptldx.dbf',
  '/private/oracle/oradata/LDAP/ptllog.dbf',
  '/private/oracle/oradata/LDAP/oca.dbf',
  '/private/oracle/oradata/LDAP/discopltc1.dbf',
  '/private/oracle/oradata/LDAP/discopltm1.dbf',
  '/private/oracle/oradata/LDAP/oss_sys01.dbf',
  '/private/oracle/oradata/LDAP/wcrsys01.dbf',
  '/private/oracle/oradata/LDAP/uddsys01.dbf',
  '/private/oracle/oradata/LDAP/b2b_dt.dbf',
  '/private/oracle/oradata/LDAP/b2b_rt.dbf',
  '/private/oracle/oradata/LDAP/b2b_idx.dbf',
  '/private/oracle/oradata/LDAP/b2b_lob.dbf',
  '/private/oracle/oradata/LDAP/bam.dbf',
  '/private/oracle/oradata/LDAP/orabpel.dbf',
  '/private/oracle/oradata/LDAP/attrs1_oid.dbf',
  '/private/oracle/oradata/LDAP/battrs1_oid.dbf',
  '/private/oracle/oradata/LDAP/gcats1_oid.dbf',
  '/private/oracle/oradata/LDAP/gdefault1_oid.dbf',
  '/private/oracle/oradata/LDAP/svrmg1_oid.dbf',
  '/private/oracle/oradata/LDAP/ias_meta01.dbf',
  '/private/oracle/oradata/LDAP/undotbs.dbf'
CHARACTER SET AL32UTF8
;

8. Edit the file newdb.sql, as follows:
a. Change the line:

```sql
CREATE CONTROLFILE REUSE DATABASE "LDAP" RESETLOGS NOARCHIVELOG
```

to say:

```sql
CREATE CONTROLFILE REUSE SET DATABASE "NLADAP" RESETLOGS NOARCHIVELOG
```

b. Modify the UNIX directory location of the database and logfiles to point to the new node site's directory.

In our example, after these modifications, `newdb.sql` should look like this:

```sql
STARTUP NOMOUNT
CREATE CONTROLFILE REUSE SET DATABASE "NLADAP" RESETLOGS NOARCHIVELOG
  MAXLOGFILES 16
  MAXLOGMEMBERS 3
  MAXDATAFILES 100
  MAXINSTANCES 8
  MAXLOGHISTORY 454
LOGFILE
  GROUP 1 '/private1/oracle/oradata/NLDAP/redo01.log'  SIZE 10M,
  GROUP 2 '/private1/oracle/oradata/NLDAP/redo02.log'  SIZE 10M,
  GROUP 3 '/private1/oracle/oradata/NLDAP/redo03.log'  SIZE 10M
-- STANDBY LOGFILE
DATAFILE
  '/private1/oracle/oradata/NLDAP/system01.dbf',
  '/private1/oracle/oradata/NLDAP/sysaux01.dbf',
  '/private1/oracle/oradata/NLDAP/users01.dbf',
  '/private1/oracle/oradata/NLDAP/dcm.dbf',
  '/private1/oracle/oradata/NLDAP/portal.dbf',
  '/private1/oracle/oradata/NLDAP/ptldoc.dbf',
  '/private1/oracle/oradata/NLDAP/ptlidx.dbf',
  '/private1/oracle/oradata/NLDAP/ptllog.dbf',
  '/private1/oracle/oradata/NLDAP/oca.dbf',
  '/private1/oracle/oradata/NLDAP/discopltc1.dbf',
  '/private1/oracle/oradata/NLDAP/discopltm1.dbf',
  '/private1/oracle/oradata/NLDAP/css_sys01.dbf',
  '/private1/oracle/oradata/NLDAP/wcrsys01.dbf',
  '/private1/oracle/oradata/NLDAP/uddisys01.dbf',
  '/private1/oracle/oradata/NLDAP/b2b_dt.dbf',
  '/private1/oracle/oradata/NLDAP/b2b_rt.dbf',
  '/private1/oracle/oradata/NLDAP/b2b_idx.dbf',
  '/private1/oracle/oradata/NLDAP/b2b_lob.dbf',
  '/private1/oracle/oradata/NLDAP/bam.dbf',
  '/private1/oracle/oradata/NLDAP/orabpel.dbf',
  '/private1/oracle/oradata/NLDAP/attrs1_oid.dbf',
  '/private1/oracle/oradata/NLDAP/battrs1_oid.dbf',
  '/private1/oracle/oradata/NLDAP/gcats1_oid.dbf',
  '/private1/oracle/oradata/NLDAP/gdefault1_oid.dbf',
  '/private1/oracle/oradata/NLDAP/svrmgl_oid.dbf',
  '/private1/oracle/oradata/NLDAP/ias_meta01.dbf',
  '/private1/oracle/oradata/NLDAP/undotbs.dbf'
CHARACTER SET AL32UTF8
;
```

9. Copy the initialization parameter file `init$ORACLE_SID.ora` of the sponsor directory's database to `init$ORACLE_SID_NEW_DIR_DB.ora`. The default location of the initialization parameter file is `$ORACLE_HOME/dbs` on UNIX and `%ORACLE_HOME%\database` on Windows. In our example, copy

```
/private/oracle/app/oracle/product/OraHome_1/dbs/initLDAP.ora
```
Addition of an Oracle Advanced Database Replication-Based Directory Node

Addition of a Directory Node by Using the Database Copy Procedure

I-7

to /private/oracle/app/oracle/product/OraHome_1/dbs/initNLDAP.ora as shown here:

```bash
$cd $ORACLE_HOME/dbs
$cp initLDAP.ora initNLDAP.ora
```

If you are using the server parameter file `spfile$ORACLE_SID.ora` or `spfile.ora` instead of an initialization parameter file, create an initialization parameter file from the server parameter file as shown in the following example:

```bash
$sqlplus /nolog
SQL> connect / as sysdba
SQL> create pfile from spfile
```

The previous example assumes that `spfile$ORACLE_SID.ora` is located in the default location `$ORACLE_HOME/dbs`. In our example, the previous step will create an `initLDAP.ora` file at 
/private/oracle/app/oracle/product/OraHome_1 from `spfileLDAP.ora`. If the server parameter file is not located in the default location, you must include the complete path, as shown in the following example:

```bash
$sqlplus /nolog
SQL> connect / as sysdba
SQL> create pfile='/private/oracle/initLDAP.ora' from
spfile='/private/oracle/initLDAP.ora'
```

After you create the initialization file parameter file, create a copy of it as explained at the beginning of this step.

10. In the new initialization parameter file, make following changes:

   a. Comment the parameter `JOB_QUEUE_PROCESSES`.
   
   b. Change the parameter `dbname` from `LDAP` to `NLDAP`.
   
   c. If the new site's domain name is different from the sponsor directory's domain name, alter the parameter `db_domain` also.
   
   d. Alter the location of following parameters to point to location of the new site.

```
background_dump_dest
core_dump_dest
user_dump_dest
control_files
db_recovery_file_dest
```

   e. In addition to the parameters listed in step c, if your initialization parameter file has any parameters that are node specific, such as `DB_RECOVERY_FILE_DEST` and `DB_CREATE_FILE_DEST`, alter those parameters as well.

In our example, the initialization parameter file `initNLDAP.ora` will look like this after these modifications:

```
*.aq_tm_processes=1
*.background_dump_dest='/privat1/oracle/app/oracle/admin/NLDAP/bdump'
*.compatible='10.1.0.2.0'
*.control_files='/privat1/oracle/app/oracle/admin/NLDAP/control01.ctl',
   '/privat1/oracle/app/oracle/admin/NLDAP/control02.ctl',
   '/privat1/oracle/app/oracle/admin/NLDAP/control03.ctl'
*.core_dump_dest='/privat1/oracle/app/oracle/admin/NLDAP/cdump'
*.db_block_size=8192
*.db_cache_size=50331648
*.db_domain='acme.com'
```
*.db_file_multiblock_read_count=16
*.db_name='NLDAP'
*.db_recovery_file_dest='/private/oracle1/app/oracle/flash_recovery_area'
*.db_recovery_file_dest_size=2147483648
*.dispatchers='(PROTOCOL=TCP)(PRE=oracle.aurora.server.GiopServer)' ,
' (PROTOCOL=TCP)(PRE=oracle.aurora.server.SGiopServer)'
*.java_pool_size=67108864
*.job_queue_processes=5
*.large_pool_size=8388608
*.max_commit_propagation_delay=0
*.open_cursors=300
*.pga_aggregate_target=33554432
*.processes=150
*.remote_login_passwordfile='EXCLUSIVE'
*.sessions=400
*.shared_pool_size=150994944
*.undo_management='AUTO'
*.undo_tablespace='UNDOTBS'
*.user_dump_dest='/private1/oracle/app/oracle/admin/NLDAP/udump'

11. Edit the tnsnames.ora file to include information connection details of to the new node. Refer to the following sample file:

```
LDAP.ACME.COM =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = rst-sun)(PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = ldap.acme.com)
  )
)
NLDAP.ACME.COM =
(DESCRIPTION =
  (ADDRESS = (PROTOCOL = TCP)(HOST = dsm-sun)(PORT = 1521))
  (CONNECT_DATA =
    (SERVER = DEDICATED)
    (SERVICE_NAME = nldap.acme.com)
  )
)
```

12. Copy the file listener.ora to list.bak. Edit the copied file list.bak to include the information pertaining to the new node. Refer to the following sample file:

```
# The KEY value for the IPC protocol may be anything, and
# is not related to either the TCP hostname or database SID.
LISTENER =
  (ADDRESS_LIST =
    (ADDRESS=(PROTOCOL= IPC)(KEY= LDAP))
    (ADDRESS=(PROTOCOL= IPC)(KEY= PNPKEY))
    (ADDRESS=(PROTOCOL= TCP)(Host= dsm-sun)(Port= 1521))
  )
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME= dsm-sun.us.oracle.com)
      (ORACLE_HOME= /privatel/oracle/app/oracle/product/OraHome_1)
      (SID_NAME = NLDAP)
    )
  )
```
Addition of an Oracle Advanced Database Replication-Based Directory Node

Addition of a Directory Node by Using the Database Copy Procedure

SID_NAME = extproc
(ORACLE_HOME = /privat1/oracle/app/oracle/product/OraHome_1)
(PROGRAM = extproc)
)
)
STARTUP_WAIT_TIME_LISTENER = 0
CONNECT_TIMEOUT_LISTENER = 10
TRACE_LEVEL_LISTENER = OFF

The default location of tnsnames.ora and listener.ora is the directory $ORACLE_HOME/network/admin on UNIX and the directory ORACLE_HOME\network\admin on Windows. It can also be in one of these locations:

- The directory specified by the TNS_ADMIN environment variable or registry value.
- On UNIX operating systems, the global configuration directory. For example, on Solaris, this directory is /var/opt/oracle.

13. Copy the updated tnsnames.ora file to the other nodes. You can copy them to other nodes either using FTP or another appropriate method.

Prior to copying the file tnsnames.ora to the new node, install the Oracle database software on the new node. Also copy the files list.bak, listener.ora, and sqlnet.ora from the sponsor node to the new node.

14. Create an archive of all the data files and compress the archived file. For example:

```bash
>> $ find / -name *.dbf -print \
   -exec tar rvf tar_file_name_with_absolute_path {} \;

Note: All files listed under DATAFILE in newdb.sql must be archived.
```

This command will search for all files ending with extension .dbf, starting from the root directory. The assumption is that there is only one instance of the database server installed on the node and that data files end with the .dbf extension.

Compress the archive file:

```bash
>> $ compress tar_file_name_with_absolute_path
```

This procedure is only an example to illustrate the method to back up the files. The Oracle data files will be backed up in the absolute path using this method. It is a better idea to back up the files from the current directory, so that you have more flexibility as to where you can restore the data files. Consult your system administrator before backing up the database.

Tasks To Be Performed on the New Advanced Replication Node

Complete the following steps on the new node.

1. Stop application server control, DCM, opmn, database and listener services in the new node:

```bash
$> emctl stop iasconsole
$> $ORACLE_HOME/dcm/bin/dcmctl stop
$> $ORACLE_HOME/opmn/bin/opmnctl stopall
$> sqlplus "/ as sysdba"
SQL> shutdown immediate;
```
Addition of an Oracle Advanced Database Replication-Based Directory Node

SQL> exit
$> lsnrctl [listener_name] stop

2. Copy the initialization parameter file initNLADP.ora from the sponsor node (rst-sun) to the new node under the UNIX directory $ORACLE_HOME/dbs using FTP or another appropriate tool. Ensure that the contents of the copied file initNLADP.ora are valid after copying.

3. Ensure that following files do not exist in the directory $ORACLE_HOME/dbs on UNIX or ORACLE_HOME\database in Windows:
   - spfileNLADP.ora
   - spfile.ora

   If either of these files exists, the Oracle database will use that file instead of the initNLADP.ora file you copied from sponsor node.

4. Copy the archived file created in step 14 of steps to be performed at the sponsor node procedure, using FTP or another appropriate tool. The following example uses FTP tool to copy the archived file from rst-sun.

   $ ftp
   ftp> open rst-sun
   Connected to rst-sun.us.oracle.com.
   220 rst-sun FTP server (UNIX(r) System V Release 4.0) ready.
   Name (rst-sun:oracle):
   331 Password required for oracle.
   Password:
   230 User oracle logged in.
   ftp> cd /private1/oracle/oradata/LDAP
   250 CWD command successful.
   ftp> binary
   200 Type set to I.
   ftp> mget oradb.tar.Z

   If the data files are huge (several gigabytes or terabytes) and the network bandwidth is low, then it may be a better idea to physically ship the compressed file on any media, such as tape or disk, from the sponsor to the new node.

   Extract the archived file on the new node. For example:

   $ uncompress oradb.tar.Z
   $ tar xvf oradb.tar

   Ensure that the data files are extracted to correct directory. In our example, it is /private1/oracle/oradata/NLDAP

5. Using FTP or another appropriate tool, copy the file newdb.sql you created in Step 5 of “Tasks To Be Performed on the Sponsor LDAP Replication Node” on page I-16. For example:

   $ cd /private1/oracle/app/oracle/admin/NLDAP/udump
   $ ftp
   ftp> open rst-sun
   ftp> cd /private1/oracle/app/oracle/admin/ldap/udump
   ftp> mget newdb.sql

6. At the UNIX shell prompt, set ORACLE_BASE, ORACLE_HOME and ORACLE_SID environment variables. For example (using the C shell):

   $ setenv ORACLE_BASE /private1/oracle/app/oracle
   $ setenv ORACLE_HOME /private1/oracle/app/oracle/product/OraHome_1
Addition of an Oracle Advanced Database Replication-Based Directory Node

7. In the same UNIX shell, execute `newdb.sql` using SQL*Plus as shown in the following example:

```
$ setenv ORACLE_SID NLDAP
$ sqlplus /nolog
SQL> connect /as sysdba
SQL> @newdb.sql
SQL> shutdown normal
SQL> exit
```

8. Edit the initialization parameter file `initNLDAP.ora` in the `$ORACLE_HOME/dbs` directory to uncomment the parameter `job_queue_processes`. Its value must be equal to or greater than the number of nodes in the Directory Replication Group.

9. Start up the database and listener as follows:

```
$ sqlplus /nolog
SQL> connect /as sysdba
SQL> startup mount
SQL> alter database open resetlogs
SQL> exit
$ lsnrctl start
```

10. Log in to the sponsor node and start up the database and listener on the sponsor node. In our example, the sponsor node is rst-sun.

```
$ telnet rst-sun
$ sqlplus /nolog
SQL> connect /as sysdba
SQL> startup
SQL> exit
$ lsnrctl start
$ exit
```

11. Change the global database name of the new node.

```
SQL> connect /as sysdba
SQL> alter database rename global_name to NLDAP;
SQL> exit
```

12. Add a temporary file to the tablespace using the following command:

```
SQL> connect /as sysdba
SQL> ALTER TABLESPACE TEMP ADD TEMPFILE 'temp01.dbf' size 2000k;
SQL> exit
```

13. Delete the wallet files `oidpwdlldap1` and `oidpwdr*` at new node and reset ODS password

```
$ cd $ORACLE_HOME/ldap/admin
$ rm oidpwdlldap1 oidpwdr*
```

14. Reset the password and start Oracle Internet Directory processes.

```
$ oidpasswd connect=nldap.acme.com create_wallet=true current_password=ods
$ oidmon connect=nldap.acme.com start
$ oidctl connect=nldap.acme.com server=oidldapd instance=1 start
```
15. Reset the ReplicaID at the new node. After the database copy, the replicaid in the database of the new node will be same as that of the sponsor node. Therefore, the new node's replicaid must be replaced. The new value of replicaid must be of the form hostname_sid where hostname is the host name (without the domain name) of the new node where Oracle Internet Directory server’s repository will be running and sid is the ORACLE_SID of the new node database. In our example, the replicaid is dsm-sun_nldap. Ensure that all letters of the replicaid are in lower case. To reset the value of replicaid:

a. Create a file, chgrid.ldif, with the following contents:

```
  dn:
  changetype: modify
  replace: orclreplicaid
  orclreplicaid: dsm-sun_nldap
```

b. Using the ldapmodify tool, change the replicaid:

```
$ $ORACLE_HOME/bin/ldapmodify -p port#_of_ldap_server -h new_node_hostname -f chgrid.ldif
```

16. Because the replica id of the new node was changed in step 15, you must re-create the relative replica entries for the new node, as follows:

```
$ remtool -pcleanup -bind "new_node_host:new_node_port/new_node_repl_pswd"
```

The remtool command will report an error and prompt for input because there are no replica entries corresponding to the replica id from the previous step yet. It will use your responses to rectify the error. For example,

```
remtool -pcleanup -bind "new_node_host:new_node_port/new_node_repl_pswd"
```

Error occurred while getting replication configuration information.

This tool will try to rectify the problem if super user DN and password are provided.

Do you want to continue? [y/n] : y

```
Enter superuser DN : cn=orcladmin

Enter superuser password : 

Enter new password of replication DN : 

Reenter new password of replication DN : 

DRG identified by replica ldap://new_node_host:new_node_port (new_replica_id) will be cleaned up.

Do you want to continue? [y/n] : y
```

------------------------------------------------------------------------------

------------------------------------------------------------------------------
Replica replica ldap://new_node_host:new_node_port (new_replica_id) has been cleaned up.

17. In addition to renaming the replica subentry, change the `orclreplicauri` and `orclreplicasecondaryuri` attributes of the replica subentry. The `orclreplicauri` and `orclreplicasecondaryuri` attributes must be modified to contain the URI of the new node's ldap server. Proceed as follows.

   a. Create an LDIF file, `modsubentry.ldif`, with the following contents:

      ```
      dn: orclreplicaid=new_replicaid,
      cn=replication configuration
      changetype: modify
      replace: orclreplicauri
      #Use your host name and port number
      #where ldap server is listening
      orclreplicauri: ldap://dsum-sun:389/
      -
      replace: orclreplicasecondaryuri
      #Use your fully qualified host name and
      #the port number where ldap server is listening
      orclreplicasecondaryuri: ldap://dsum-sun.acme.com:389/
      -
      replace: orclreplicastate
      orclreplicastate: 6
      ```

   b. Using the `ldapmodify` tool, apply the changes to the directory:

      ```
      $ ldapmodify -p port#_of_ldap_server -h new_node_hostname -f modsubentry.ldif
      ```

   **Note:** Set the `orclreplicastate` attribute of the configuration entry `orclreplicaid=replicaid,cn=replication configuration` to 6 at the node that will act as the Remote Master Site. The previous example assumes that the new node will be the Remote Master Site and sets the `orclreplicastate` attribute to 6 there. If you want to use the new node as the Master Definition Site and the sponsor node as the Remote Master Site, set the `orclreplicastate` attribute to 6 at the sponsor node.

18. If you have performed a database copy from a node that has Advanced replication configured with another node, you must delete the LDAP_REP replication group in the new node. To do so, execute the following command:

   ```
   $> sqlplus rep_admin_db_account_name/password@db_conn_str_of_new_node
   SQL> exec dbms_repcat.drop_master_repgroup( gname => 'LDAP_REP' )
   ```


   ```
   oidmon connect=connect_string stop
   ```

20. Clean up changelog tables at the new node.

   ```
   $ sqlplus /nolog
   SQL> connect ods/ods_password;
   SQL> truncate table ods.ods_chg_log;
   SQL> truncate table ods.ods_chg_stat;
   SQL> truncate table ods.asr_chg_log;
   ```
21. To configure Oracle Database Advanced Replication, if you are adding new node to an existing DRG, at the shell prompt, execute the following command:

   $ remtool -addnode

To configure Oracle Database Advanced Replication, if you are creating a new DRG consisting of sponsor node and new node, at the shell prompt, execute the following command:

   $ remtool -asrsetup

See Also: The remtool command-line tool reference in Oracle Identity Management User Reference

22. Start up Oracle Internet Directory and the LDAP replication server on all the nodes, including the new node and sponsor node.

   Note: The sponsor node might contain changelogs for operations that occurred before you performed the database copy procedure. If so, after the replication server is started, these changelogs will be propagated to the new node. Propagation will fail, however, because directory data in the new database copy node is already consistent with data in the sponsor node. As a result, these changelogs will end up in the Human Intervention Queue at the database copy node.

   If you have created a new DRG consisting of the sponsor node and the new node, you can avoid these errors by truncating the ods_chg_log table at the sponsor node before starting the LDAP server at the sponsor node.

   If you have added the new node to an existing DRG, do not truncate the table at sponsor node. Instead, let the changes end up in the Human Intervention Queue at the new node. Then the administrator should clean up the queue as described in "Resolving Conflicts Manually in a Replication Group" in Chapter 30.

23. Update the OIDport and OIDsslport parameters in the $ORACLE_HOME/config/ias.properties file with the non-SSL port and SSL port where the directory server of the new node is currently listening:

   [ComponentConfig]
   ...
   [InstallData]
   ...
   OIDhost=dsm-sun
   OIDport=current_non_ssl_port_of_ldap_server
   OIDsslport=current_ssl_port_of_ldap_server
   ...
   FarmAdminSupported=FALSE

Verification of an Advanced Replication-Based Replica Node

   Log in to the Oracle database by using SQL*Plus and specify the user name as ODS, and the password ods when prompted.
Check the `ods_chg_stat` table on all nodes and see if they have correct and identical rows. The `ods_chg_stat` table should contain \( \text{number of nodes} \times \text{number of nodes} \) rows. For example, if there were two nodes participating in Oracle Database Advanced Replication-based replication, and you added a third node, the `ods_chg_stat` table would contain nine rows, that is, \( 3 \times 3 \), on each node. The rows are shown in the following table:

<table>
<thead>
<tr>
<th>Supplier</th>
<th>Consumer</th>
<th>Change Number</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node1</td>
<td>node2</td>
<td>number 1</td>
</tr>
<tr>
<td>Node1</td>
<td>node3</td>
<td>number 2</td>
</tr>
<tr>
<td>Node1</td>
<td>node1</td>
<td>number 3</td>
</tr>
<tr>
<td>Node2</td>
<td>node1</td>
<td>number 4</td>
</tr>
<tr>
<td>Node2</td>
<td>node2</td>
<td>number 5</td>
</tr>
<tr>
<td>Node2</td>
<td>node2</td>
<td>number 6</td>
</tr>
<tr>
<td>Node3</td>
<td>node1</td>
<td>0</td>
</tr>
<tr>
<td>Node3</td>
<td>node2</td>
<td>0</td>
</tr>
<tr>
<td>Node3</td>
<td>node3</td>
<td>0</td>
</tr>
</tbody>
</table>

The rows with consumer names identical to that of suppliers contain the last changes processed by the outbound change log processing threads at the supplier sides. The rows with different supplier and consumer names contain last change numbers already processed from the suppliers to the consumers in question.

Since Node3 is a new node, there have been no changes supplied by Node3 yet. Therefore, the change numbers for Node3 as supplier are 0.

There may be a time delay before all nodes contain identical rows, but this delay should not be more than two to three minutes.

---

**Addition of an LDAP Replication-Based Directory Node**

This section describes the addition of a full LDAP replica directory node. The LDAP replication to this node may be either one-way or two-way. This section contains the following topics:

- **Tasks To Be Performed on the Sponsor LDAP Replication Node**
- **Tasks To Be Performed on the New LDAP Replication Node**
- **Verification of an LDAP-Based Replica Node**

**Tasks To Be Performed on the Sponsor LDAP Replication Node**

Complete the following steps on the sponsor node.

1. At the command line prompt execute SQL*Plus.
   ```
   $ sqlplus /nolog
   SQL> connect /as sysdba
   SQL> ALTER DATABASE BACKUP CONTROLFILE TO TRACE RESETLOGS;
   ```

   This command creates a trace file under the user dump destination directory. In this example, that directory is `/private/oracle/app/oracle/admin/LDAP/udump`. 
The file will be created in the following format:

$ORACLE_SID_ora_processid.trc

For example:

ldap_ora_4765.trc

2. Shut down the LDAP and replication servers and OID Monitor processes. Make sure the ldap and replication servers are stopped before stopping the OID Monitor process.

   $ oidctl connect=connect_string server=oidrepld instance=instance_number stop
   $ oidctl connect=connect_string server=oidldapd instance=instance_number stop
   $ oidmon connect=connect_string stop

   You can also stop OIDMON and the LDAP server process using opmn. Ensure that the replication server, directory server and OIDMON are shut down before you proceed to next step.

   In these commands, connect_string is the net service name in the node's tnsnames.ora file.

3. REMOVE

4. Shut down the database and Oracle Net Services listener on the sponsor node only:

   $ lsnrctl [listener_name] stop
   $ sqlplus /nolog
   SQL> connect /as sysdba
   SQL> shutdown normal
   SQL> exit

   By default, the listener name is LISTENER.

5. Copy the trace file created under Step 1 to a new file, newdb.sql, under the same directory.

   $ cd $ORACLE_BASE/admin/LDAP/udump
   $ cp ldap_ora_4765.trc newdb.sql

6. Edit newdb.sql, using any text editor, and delete all the lines except the STARTUP NOMOUNT and CREATE CONTROLFILE statements. After editing, newdb.sql should look like this:

   STARTUP NOMOUNT
   CREATE CONTROLFILE REUSE SET DATABASE "LDAP" RESETLOGS NOARCHIVELOG
   MAXLOGFILES 16
   MAXLOGMEMBERS 3
   MAXDATAFILES 100
   MAXINSTANCES 8
   MAXLOGHISTORY 454
   LOGFILE
   GROUP 1 '/private/oracle/oradata/LDAP/redo01.log' SIZE 10M,
   GROUP 2 '/private/oracle/oradata/LDAP/redo02.log' SIZE 10M,
   GROUP 3 '/private/oracle/oradata/LDAP/redo03.log' SIZE 10M
   -- STANDBY LOGFILE
   DATAFILE
   '/private/oracle/oradata/LDAP/system01.dbf',
   '/private/oracle/oradata/LDAP/sysaux01.dbf',
   '/private/oracle/oradata/LDAP/users01.dbf',
   '/private/oracle/oradata/LDAP/dcm.dbf'
Addition of a Directory Node by Using the Database Copy Procedure

7. Make following changes to newdb.sql.

a. Change the line:
   CREATE CONTROLFILE REUSE DATABASE "LDAP" RESETLOGS NOARCHIVELOG
   to say:
   CREATE CONTROLFILE REUSE SET DATABASE "NLDAP" RESETLOGS NOARCHIVELOG

b. Modify the UNIX directory location of the database and logfiles to point to the new node site's directory.

In our example, after these modifications, newdb.sql should look like this:

STARTUP NOMOUNT
CREATE CONTROLFILE REUSE SET DATABASE "NLDAP" RESETLOGS NOARCHIVELOG
   MAXLOGFILES 16
   MAXLOGMEMBERS 3
   MAXDATAFILES 100
   MAXINSTANCES 8
   MAXLOGHISTORY 454
LOGFILE
   GROUP 1 '/private1/oracle/oradata/NLDAP/redo01.log' SIZE 10M,
   GROUP 2 '/private1/oracle/oradata/NLDAP/redo02.log' SIZE 10M,
   GROUP 3 '/private1/oracle/oradata/NLDAP/redo03.log' SIZE 10M
   -- STANDBY LOGFILE
DATAFILE
   '/private1/oracle/oradata/NLDAP/system01.dbf',
   '/private1/oracle/oradata/NLDAP/sysaux01.dbf',
   '/private1/oracle/oradata/NLDAP/users01.dbf',
   '/private1/oracle/oradata/NLDAP/dcm.dbf',
   '/private1/oracle/oradata/NLDAP/portal.dbf',
   '/private1/oracle/oradata/NLDAP/ptldoc.dbf',
   '/private1/oracle/oradata/NLDAP/ptlidx.dbf',
   '/private1/oracle/oradata/NLDAP/ptllog.dbf',
   '/private1/oracle/oradata/NLDAP/orabpel.dbf',
   '/private1/oracle/oradata/NLDAP/attrs1_oid.dbf',
   '/private1/oracle/oradata/NLDAP/battrs1_oid.dbf',
   '/private1/oracle/oradata/NLDAP/gcats1_oid.dbf',
   '/private1/oracle/oradata/NLDAP/gdefault1_oid.dbf',
   '/private1/oracle/oradata/NLDAP/svrmg1_oid.dbf',
   '/private1/oracle/oradata/NLDAP/ias_meta01.dbf',
   '/private1/oracle/oradata/NLDAP/undotbs.dbf'
Addition of an LDAP Replication-Based Directory Node

'/private1/oracle/oradata/NLDAP/oca.dbf',
'/private1/oracle/oradata/NLDAP/discopltc1.dbf',
'/private1/oracle/oradata/NLDAP/discopltm1.dbf',
'/private1/oracle/oradata/NLDAP/oss_sys01.dbf',
'/private1/oracle/oradata/NLDAP/wcrsys01.dbf',
'/private1/oracle/oradata/NLDAP/uddisys01.dbf',
'/private1/oracle/oradata/NLDAP/b2b_dt.dbf',
'/private1/oracle/oradata/NLDAP/b2b_rt.dbf',
'/private1/oracle/oradata/NLDAP/b2b_idx.dbf',
'/private1/oracle/oradata/NLDAP/b2b_lob.dbf',
'/private1/oracle/oradata/NLDAP/bam.dbf',
'/private1/oracle/oradata/NLDAP/orabpel.dbf',
'/private1/oracle/oradata/NLDAP/attrs1_oid.dbf',
'/private1/oracle/oradata/NLDAP/battrs1_oid.dbf',
'/private1/oracle/oradata/NLDAP/gcats1_oid.dbf',
'/private1/oracle/oradata/NLDAP/gdefault1_oid.dbf',
'/private1/oracle/oradata/NLDAP/svrmg1_oid.dbf',
'/private1/oracle/oradata/NLDAP/ias_meta01.dbf',
'/private1/oracle/oradata/NLDAP/undotbs.dbf'
CHARACTER SET AL32UTF8

8. Copy the initialization parameter file init$ORACLE_SID.ora of the sponsor directory’s database to init$ORACLE_SID_NEW_DIR_DB.ora. The default location of the initialization parameter file is $ORACLE_HOME/dbs on UNIX and %ORACLE_HOME%\database on Windows. In our example, copy /private/oracle/app/oracle/product/OraHome_1/dbs/initLDAP.ora to /private/oracle/app/oracle/product/OraHome_1/dbs/initNLDAP.ora as shown here:

$cd $ORACLE_HOME/dbs
$cp initLDAP.ora initNLDAP.ora

If you are using the server parameter file spfile$ORACLE_SID.ora or spfile.ora instead of an initialization parameter file, create an initialization parameter file from the server parameter file as shown in the following example:

$sqlplus /nolog
SQL> connect / as sysdba
SQL> create pfile from spfile

The previous example assumes that spfile$ORACLE_SID.ora is located in the default location $ORACLE_HOME/dbs. In our example, the previous step will create an initLDAP.ora file at /private/oracle/app/oracle/product/OraHome_1 from spfileLDAP.ora. If the server parameter file is not located in the default location, you must include the complete path, as shown in the following example:

$sqlplus /nolog
SQL> connect / as sysdba
SQL> create pfile="/private/oracle/initLDAP.ora" from
spfile="/private/oracle/initLDAP.ora"

After you create the initialization file parameter file, create a copy of it as explained at the beginning of this step.

9. In the new initialization parameter file, make following changes:

a. Change the parameter dbname from LDAP to NLDAP.
b. If the new site’s domain name is different from the sponsor directory’s domain name, alter the parameter `db_domain` also.

c. Alter the location of following parameters to point to location of the new site.

```plaintext
background_dump_dest
core_dump_dest
user_dump_dest
control_files
db_recovery_file_dest
```

d. In addition to the parameters listed in step c, if your initialization parameter file has any parameters that are node specific, such as `DB_RECOVERY_FILE_DEST` and `DB_CREATE_FILE_DEST`, alter those parameters as well.

In our example, the initialization parameter file `initNLDAP.ora` will look like this after these modifications:

```plaintext
*.aq_tm_processes=1
*.background_dump_dest='/private1/oracle/app/oracle/admin/NLDAP/bdump'
*.compatible='10.1.0.2.0'
*.control_files='/private1/oracle/app/oracle/admin/NLDAP/control01.ctl',
   '/private1/oracle/app/oracle/admin/NLDAP/control02.ctl',
   '/private1/oracle/app/oracle/admin/NLDAP/control03.ctl'
*.core_dump_dest='/private1/oracle/app/oracle/admin/NLDAP/cdump'
*.db_block_size=8192
*.db_cache_size=50331648
*.db_domain='acme.com'
*.db_file_multiblock_read_count=16
*.db_name='NLDAP'
*.db_recovery_file_dest='private/oracle1/app/oracle/flash_recovery_area'
*.db_recovery_file_dest_size=2147483648
*.dispatchers='(PROTOCOL=TCP) (PRE=oracle.aurora.server.GiopServer)',
   '(PROTOCOL=TCP) (PRE=oracle.aurora.server.SGiopServer)'*
*.java_pool_size=67108864
#$*.job_queue_processes=5
*.large_pool_size=8388608
*.max_commit_propagation_delay=0
*.open_cursors=300
*.pga_aggregate_target=33554432
*.processes=150
*.remote_login_passwordfile='EXCLUSIVE'
*.sessions=400
*.shared_pool_size=150994944
*.undo_management='AUTO'
*.undo_tablespace='UNDOTBS'
*.user_dump_dest='/private1/oracle/app/oracle/admin/NLDAP/udump'
```

10. Create an archive of all the data files and compress the archived file. For example:

```bash
$ find / -name *.dbf -print \
-exec tar rvf tar_file_name_with_absolute_path {} \;
```

**Note:** All files listed under DATAFILE in `newdb.sql` must be archived.

This command will search for all files ending with extension `.dbf`, starting from the root directory. The assumption is that there is only one instance of the database server installed on the node and that data files end with the `.dbf` extension.
Compress the archive file:

```
$ compress tar_file_name_with_absolute_path
```

This procedure is only an example to illustrate the method to back up the files. The Oracle data files will be backed up in the absolute path using this method. It is a better idea to back up the files from the current directory, so that you have more flexibility as to where you can restore the data files. Consult your system administrator before backing up the database.

**Tasks To Be Performed on the New LDAP Replication Node**

Complete the following steps on the new node.

1. Stop application server control, DCM, opmn, database and listener services in the new node:
   ```
   $> emctl stop iasconsole
   $> $ORACLE_HOME/dcm/bin/dcmct1 stop
   $> $ORACLE_HOME/opmn/bin/opmnctl stopall
   $> sqlplus "/ as sysdba"
   SQL> shutdown immediate;
   SQL> exit
   $> lsnrctl [listener_name] stop
   ```

2. Copy the initialization parameter file `initNLDAP.ora` from the sponsor node (rst-sun) to the new node under the UNIX directory `$ORACLE_HOME/dbs` using FTP or another appropriate tool. Ensure that the contents of the copied file `initNLDAP.ora` are valid after copying.

3. Ensure that following files do not exist in the directory `$ORACLE_HOME/dbs` on UNIX or `ORACLE_HOME\database` in Windows:
   - `spfileNLDAP.ora`
   - `spfile.ora`

   If either of these files exists, the Oracle database will use that file instead of the `initNLDAP.ora` file you copied from sponsor node.

4. Copy the archived file created in step 14 of steps to be performed at the sponsor node procedure, using FTP or another appropriate tool. The following example uses FTP tool to copy the archived file from rst_sun.
   ```
   $ ftp
   ftp> open rst-sun
   Connected to rst-sun.us.oracle.com.
   220 rst-sun FTP server (UNIX(r) System V Release 4.0) ready.
   Name (rst-sun:oracle):
   331 Password required for oracle.
   Password:
   230 User oracle logged in.
   ftp> cd /private1/oracle/oradata/LDAP
   250 CWD command successful.
   ftp> binary
   200 Type set to I.
   ftp> mget oradb.tar.Z
   ```

   If the data files are huge (several gigabytes or terabytes) and the network bandwidth is low, then it may be a better idea to physically ship the compressed file on any media, such as tape or disk, from the sponsor to the new node.
Extract the archived file on the new node. For example:

```bash
$ uncompress oradb.tar.Z
$ tar xvf oradb.tar
```

Ensure that the data files are extracted to correct directory. In our example, it is `/private1/oracle/oradata/NLDAP`

5. Using FTP or another appropriate tool, copy the file `newdb.sql` you created in Step 5 of “Tasks To Be Performed on the Sponsor LDAP Replication Node” on page I-16. For example:

```bash
$ cd /private1/oracle/app/oracle/admin/NLDAP/udump
$ ftp
ftp> open rst-sun
ftp> cd /private1/oracle/app/oracle/admin/LDAP/udump
ftp> mget newdb.sql
```

6. At the UNIX shell prompt, set ORACLE_BASE, ORACLE_HOME and ORACLE_SID environment variables. For example (using the C shell):

```bash
$ setenv ORACLE_BASE /private1/oracle/app/oracle
$ setenv ORACLE_HOME /private1/oracle/app/oracle/product/OraHome_1
$ setenv ORACLE_SID NLDAP
```

7. In the same UNIX shell, execute `newdb.sql` using SQL*Plus as shown in the following example:

```sql
$ sqlplus /nolog
SQL> connect /as sysdba
SQL> @newdb.sql
SQL> shutdown normal
SQL> exit
```

8. Edit the initialization parameter file `initNLDAP.ora` in the `$ORACLE_HOME/dbs` directory to uncomment the parameter `job_queue_processes`. Its value must be equal to or greater than the number of nodes in the Directory Replication Group.

9. Start up the database and listener as follows:

```bash
$ sqlplus /nolog
SQL> connect /as sysdba
SQL> startup mount
SQL> alter database open resetlogs
SQL> exit
$ lsnrctl start
```

10. Log in to the sponsor node and start up the database and listener on the sponsor node. In our example, the sponsor node is rst-sun.

```bash
$ telnet rst-sun
$ sqlplus /nolog
SQL> connect /as sysdba
SQL> startup
SQL> exit
$ lsnrctl start
$ exit
```

11. Change the global database name of the new node.

```sql
SQL> connect /as sysdba
SQL> alter database rename global_name to NLDAP;
```
SQL> exit

12. Add a temporary file to the tablespace using the following command:

   SQL> connect /as sysdba
   SQL> ALTER TABLESPACE TEMP ADD TEMPFILE 'temp01.dbf' size 2000k;
   SQL> exit

13. Delete the wallet files oidpwdlldap1 and oidpwdr* at new node and reset ODS password

   $ cd $ORACLE_HOME/ldap/admin
   $ rm oidpwdlldap1 oidpwdr*

14. Reset the password and start Oracle Internet Directory processes.

   $ oidpasswd connect=nldap.acme.com create_wallet=true current_password=ods
   $ oidmon connect=nldap.acme.com start
   $ oidctl connect=nldap.acme.com server=oidldapd instance=1 start

15. Reset the ReplicaID at the new node. After the database copy, the replicaid in the database of the new node will be same as that of the sponsor node. Therefore, the new node’s replicaid must be replaced. The new value of replicaid must be of the form hostname_sid where hostname is the host name (without the domain name) of the new node where Oracle Internet Directory server's repository will be running and sid is the ORACLE_SID of the new node database. In our example, the replicaid is dsm-sun_nldap. Ensure that all letters of the replicaid are in lower case. To reset the value of replicaid:

   a. Create a file, chgrid.ldif, with the following contents:

      dn:
      changetype: modify
      replace: orclreplicaid
      orclreplicaid: dsm-sun_nldap

   b. Using the ldapmodify tool, change the replicaid:

      $ $ORACLE_HOME/bin/ldapmodify -p port#_of_ldap_server -h new_node_hostname
      \  
      -f chgrid.ldif

16. Because the replica id of the new node was changed in step 15, you must re-create the relative replica entries for the new node, as follows:

   $ remtool -pcleanup -bind "new_node_host:new_node_port/new_node_repl_pswd"

   The remtool command will report an error and prompt for input because there are no replica entries corresponding to the replica id from the previous step yet. It will use your responses to rectify the error. For example,

   remtool -pcleanup -bind "new_node_host:new_node_port/new_node_repl_pswd"

   Error occurred while getting replication configuration information.

   This tool will try to rectify the problem if super user DN and password are provided.

   Do you want to continue? [y/n] : y
Addition of an LDAP Replication-Based Directory Node

Enter superuser DN : cn=orcladmin

Enter superuser password :

Enter new password of replication DN :

Reenter new password of replication DN :

DRG identified by replica ldap://new_node_host:new_node_port (new_replica_id) will be cleaned up.

Do you want to continue? [y/n] : y

Replica replica ldap://new_node_host:new_node_port (new_replica_id) has been cleaned up.

17. In addition to renaming the replica subentry, change the orclreplicauri, orclreplicasecondaryuri and orclreplicastate attributes of the replica subentry. The orclreplicauri and orclreplicasecondaryuri attributes must be modified to contain the URI of the new node’s ldap server. The orclreplicastate attribute must be set to 6, which will be used by remtool to identify that this a database copy-based addnode. To change the values, proceed as follows.

a. Create an LDIF file, modsubentry.ldif, with the following contents:

    dn: orclreplicaid=new_replicaid,
    cn=replication configuration
    changetype: modify
    replace: orclreplicauri
    #Use your host name and port number
    #where ldap server is listening
    orclreplicauri: ldap://dsum-sun:389/
    replace:orclreplicasecondaryuri
    #Use your fully qualified host name and
    #the port number where ldap server is listening
    orclreplicasecondaryuri:
    ldap://dsum-sun.acme.com:389/

b. Using the ldapmodify tool, apply the changes to the directory:

    $ ldapmodify -p port#_of_ldap_server -h new_node_hostname -f modsubentry.ldif

18. If you have performed a database copy from a node that has Advanced replication configured with another node, you must delete the LDAP_REP replication group in the new node. To do so, execute the following command

    sqlplus rep_admin_db_account_name/password
    SQL> exec dbms_repcat.drop_master_repgroup( gname => ‘LDAP_REP’ )

oidmon connect=connect_string stop

20. Clean up changelog tables at the new node.

$ sqlplus /nolog
SQL> connect ods/ods_password;
SQL> truncate table ods.ods_chg_log;
SQL> truncate table ods.ods_chg_stat;
SQL> truncate table ods.asr_chg_log;

21. Configure LDAP Replication and add the full replica as fan-out, as follows:

a. Make sure that the database and Oracle Internet Directory server is running at the sponsor node. On the sponsor node, type:

$ sqlplus /nolog
SQL> connect /as sysdba
SQL> startup
SQL> exit
$ lsnrctl start

$ oidmon connect=sponsor_connect_string start
$ oidctl server=oidldapd inst=1 connect=sponsor_connect_string flags="-p sponsor_node_port " start

b. Make sure that Oracle Internet Directory server is running at sponsor node. On the new node, type:

$ oidmon connect=new_connect_string start
$ oidctl server=oidldapd inst=1 connect=new_connect_string flags="p new_node_port " start

c. Configure LDAP-based replication using remtool, as follows:

remtool -paddnode

See Also: The remtool command-line tool reference in Oracle Identity Management User Reference

22. Initialize replication change status of the new replication agreement.

a. Get the maximum used change number from the sponsor node:

$ ldapsearch -h sponsor_node_host -p sponsor_node_port -b " " -s base "objectclass=" lastchangenumber

b. Create an LDIF file chgstatus.ldif, with the following contents:

dn: orclagreementid=new_agreement_id, orclreplicaid=new_replica_id, cn=replication configuration
changetype:modify
replace: orcllastappliedchangenumber
orcllastappliedchangenumber: Number_from_step_a

c. Using ldapmodify, apply the change to both the sponsor node and the new node:

ldapmodify -p sponsor_node_port -h sponsor_node_host -v -f chgstatus.ldif
ldapmodify -p new_node_port -h new_node_host -v -f chgstatus.ldif
23. Start up Oracle Internet Directory and the LDAP replication server on all the nodes. For one-way replication, you only need to start replication on the consumer node.

**Note:** The sponsor node might contain changelogs for operations that occurred before you performed the database copy procedure. If so, after the replication server is started, these changelogs will be propagated to the new node. Propagation will fail, however, because directory data in the new database copy node is already consistent with data in the sponsor node. As a result, these changelogs will end up in the Human Intervention Queue at the database copy node.

If you have created a new DRG consisting of the sponsor node and the new node, you can avoid these errors by truncating the ods_chg_log table at the sponsor node before starting the LDAP server at the sponsor node.

If you have added the new node to an existing DRG, do not truncate the table at sponsor node. Instead, let the changes end up in the Human Intervention Queue at the new node. Then the administrator should clean up the queue as described in "Resolving Conflicts Manually in a Replication Group" in Chapter 30.

24. Update the OIDport and OIDsslport parameters in the $ORACLE_HOME/config/ias.properties file with the non-SSL port and SSL port where the directory server of the new node is currently listening:

```
[ComponentConfig]
...
[InstallData]
...
OIDhost=dsm-sun
OIDport=current_non_ssl_port_of_ldap_server
OIDsslport=current_ssl_port_of_ldap_server
...
FarmAdminSupported=FALSE
```

**Verification of an LDAP-Based Replica Node**

Use Oracle Directory Manager to verify that the directory replication servers are running, and then test directory replication by doing the following:

1. Log in to Oracle Directory Manager as orcladmin.

2. In the navigator pane, expand in succession Oracle Internet Directory Servers, directory server instance, Entry Management.

3. Create a single entry on the sponsor node.

   The identical entry appears in approximately 1 to 10 minutes on the consumer node. You can adjust the timing in the replication server configuration set entry.
You can use Oracle Internet Directory as a centralized directory for user authentication and authorization in a UNIX or Linux environment. The advantages of doing so include:

- Oracle Internet Directory replaces the traditional `/etc/passwd` file or Network Information System/Yellow Pages (NIS/YP)
- Password policy enforcement is easier
- Users can be classified into multiple privilege groups and managed with different privileges, depending on the server or service that they are using.

The Oracle White Paper *Centralizing UNIX Authentication and User Provisioning with Oracle Internet Directory* describes, in detail, the steps required to implement this solution. That document provides nearly complete information on using Oracle Internet Directory with Pluggable Authentication Modules (PAM). You should, however, read this appendix in addition to the white paper. This appendix includes the following sections:

- Schema Customization
- UID Attribute Issues

### Schema Customization

In 10g Release 2 (10.1.2) and later releases, you need not customize the schema as described in the white paper. The necessary attributes and object classes are available in a standard Oracle Internet Directory installation. One potential exception is the customized login attribute. You might need to add a custom login attribute to the schema, as explained in the next section.

### UID Attribute Issues

By default, Oracle products, such as OracleAS Portal, use the Oracle Internet Directory attribute `uid` for authentication and authorization. Also by default, UNIX-based operating systems and PAM use the attribute `uid` for authentication and authorization. Unfortunately, Oracle and UNIX have different requirements for acceptable `uid` strings. For example, the email address, `user@address` is a common `uid` format in Oracle Internet Directory installations. UNIX, however, does not allow the `@` character in a `uid`. There are two ways to deal with this discrepancy:

- Use a custom login attribute for PAM authentication and authorization. This requires a modification to the Oracle Internet Directory schema, as well as...
modifications to PAM. See the Oracle White Paper Centralizing UNIX Authentication and User Provisioning with Oracle Internet Directory for more information.

See Also:  "Attributes in the Directory" on page 11-8.

- Use the Oracle Internet Directory uid attribute with UNIX, but choose an alternative format to the email address. For example, you could use a short user name such as jsmith.
Table K-1 contains a list of the RFCs supported by Oracle Internet Directory.

<table>
<thead>
<tr>
<th>Number</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 1274</td>
<td>The COSINE and Internet X.500 Schema</td>
</tr>
<tr>
<td>RFC 1488</td>
<td>The X.500 String Representation of Standard Attribute Syntaxes</td>
</tr>
<tr>
<td>RFC 1777</td>
<td>Lightweight Directory Access Protocol</td>
</tr>
<tr>
<td>RFC 1778</td>
<td>The String Representation of Standard Attribute Syntaxes</td>
</tr>
<tr>
<td>RFC 1779</td>
<td>A String Representation of Distinguished Names</td>
</tr>
<tr>
<td>RFC 1960</td>
<td>A String Representation of LDAP Search Filters</td>
</tr>
<tr>
<td>RFC 2079</td>
<td>Definition of an X.500 Attribute Type and an Object Class to Hold Uniform Resource Identifiers (URIs)</td>
</tr>
<tr>
<td>RFC 2247</td>
<td>Using Domains in LDAP/X.500 Distinguished Names</td>
</tr>
<tr>
<td>RFC 2251</td>
<td>Lightweight Directory Access Protocol (v3)</td>
</tr>
<tr>
<td>RFC 2252</td>
<td>Lightweight Directory Access Protocol (v3) Attribute Syntax Definitions</td>
</tr>
<tr>
<td>RFC 2253</td>
<td>Lightweight Directory Access Protocol (v3) UTF-8 String Representation of Distinguished Names</td>
</tr>
<tr>
<td>RFC 2254</td>
<td>The String Representation of LDAP Search Filters</td>
</tr>
<tr>
<td>RFC 2255</td>
<td>The LDAP URL Format</td>
</tr>
<tr>
<td>RFC 2256</td>
<td>A summary of the X500 (96) User Schema for use with LDAPv3</td>
</tr>
<tr>
<td>RFC 2307</td>
<td>An Approach for Using LDAP as a Network Information Service</td>
</tr>
<tr>
<td>RFC 2377</td>
<td>Naming Plan for Internet Directory-Enabled Applications</td>
</tr>
<tr>
<td>RFC 2396</td>
<td>Uniform Resource Identifiers (URI): Generic Syntax</td>
</tr>
<tr>
<td>RFC 2596</td>
<td>Use of Language Codes in LDAP</td>
</tr>
<tr>
<td>RFC 2696</td>
<td>LDAP Control Extension for Simple Paged Results Manipulation</td>
</tr>
<tr>
<td>RFC 2713</td>
<td>Schema for Representing Java(tm) Objects in an LDAP Directory</td>
</tr>
<tr>
<td>RFC 2798</td>
<td>Definition of the inetOrgPerson LDAP Object Class</td>
</tr>
<tr>
<td>RFC 2829</td>
<td>Authentication Methods for LDAP</td>
</tr>
</tbody>
</table>
### Table K–1  (Cont.) Supported RFCs

<table>
<thead>
<tr>
<th>Number</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>RFC 2849</td>
<td>The LDAP Data Interchange Format (LDIF) - Technical Specification</td>
</tr>
<tr>
<td>RFC 2891</td>
<td>LDAP Control Extension for Server Side Sorting of Search Results</td>
</tr>
<tr>
<td>RFC 3112</td>
<td>LDAP Authentication Password Schema</td>
</tr>
<tr>
<td>RFC 3296</td>
<td>Named Subordinate References in Lightweight Directory Access Protocol (LDAP) Directories</td>
</tr>
<tr>
<td>RFC 3377</td>
<td>Core LDAP Requirements</td>
</tr>
<tr>
<td>RFC 3671</td>
<td>Collective Attributes in the Lightweight Directory Access Protocol (LDAP)</td>
</tr>
</tbody>
</table>

**See Also:** The Internet Engineering Task Force home page at [http://www.ietf.org/](http://www.ietf.org/).
Troubleshooting Oracle Internet Directory

This appendix explains typical problems that you could encounter while running or installing Oracle Internet Directory. It contains these sections:

- Problems and Solutions
- Need More Help?

Problems and Solutions

This section describes common Oracle Internet Directory error messages, problems and solutions. It contains the following topics:

- Installation Errors
- TCP/IP Problems
- Directory Server Error Messages and Causes
- Troubleshooting Password Policies
- Troubleshooting Directory Performance
- Troubleshooting Starting, Stopping, and Restarting of the Directory Server
- Troubleshooting Oracle Internet Directory Replication
- Troubleshooting SSL Setup
- Troubleshooting Change Log Garbage Collection
- Troubleshooting Dynamic Password Verifiers
- Troubleshooting Oracle Internet Directory Password Wallets
- Troubleshooting bulkload

Installation Errors

During installation and configuration of the Oracle Database, Oracle recommends that you select the character set UTF-8 to avoid possible problems with multibyte characters.

TCP/IP Problems

TCP/IP bugs in the operating system can interfere with Oracle Internet Directory service.
Do Not Use TCP-Based Monitoring of Oracle Internet Directory Server Availability on Microsoft Windows 2003 Server

If you use the F5 load balancer for monitoring Oracle Internet Directory server availability, configure the load balancer to use LDAP- or HTTP-based monitoring, as described in the Oracle Application Server High Availability Guide section "Configuring A Load Balancer For OracleAS Cluster (Identity Management)." Using TCP-based monitoring might cause the service to become unavailable, due to an operating system bug on Windows 2003 Server.

Do Not Install DaimondCS Port Explorer

Oracle Internet Directory will not work if DaimondCS Port Explorer is installed on the system.

Directory Server Error Messages and Causes

This section contains a list of all the Oracle directory server error messages that you can encounter. Each message is followed by its most probable causes.

Oracle Database Server Error Due to Interrupted Client Connection

You get error sgslunrRead or 30SendPort

Problem

These errors indicates that an LDAP client has disconnected abruptly.

Possible reasons include:

 ■ The client program terminated the connection without performing an unbind or abandon.
    The client machine shut down.

 ■ A network component, such as a load balancer or firewall, broke the connection due to a configured timeout setting.

 ■ The network is down.

Solution

These errors are due to conditions external to the server. If necessary, inform the network administrator.

Oracle Database Server Error Due to Schema Modifications

You get error ORA-1562.

Problem

If you attempt to add more schema components than can fit in the rollback segment space, you will encounter this error and the modifications will not commit.

Solution

To solve this, increase the size of the rollback segments in the database server.

Constraint Violation Error Due to Editing a User or Group or Creating a Realm

You get the following error in oidldap*.log:

ORA-01483: invalid length for DATE or NUMBER bind variable.
You may also see the following error on your screen:

LDAP: error code 19 - Constraint Violation

These errors might only occur intermittently.

**Problem**

If you loaded the OracleAS Metadata Repository into an Oracle 10g Database that uses the AL32UTF8 character set, you may encounter some errors when you try to edit a user or Group, or Create Identity Management Realms in Oracle Internet Directory. Editing a user includes editing attributes for an existing user.

**Solution**

As a workaround, you can wait a bit and try editing the user again.

**Standard Error Messages Returned from Oracle Directory Server**

Table L–1 lists standard error messages and their causes. Oracle Internet Directory also returns other messages listed and described in "Additional Directory Server Error Messages" on page L-5.

### Table L–1 Standard Error Messages

<table>
<thead>
<tr>
<th>Error</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>00: LDAP_SUCCESS</td>
<td>The operation was successful.</td>
</tr>
<tr>
<td>01: LDAP_OPERATIONS_</td>
<td>General errors encountered by the server when processing the</td>
</tr>
<tr>
<td>ERROR</td>
<td>request.</td>
</tr>
<tr>
<td>02: LDAP_PROTOCOL_</td>
<td>The client request did not meet the LDAP protocol requirements,</td>
</tr>
<tr>
<td>ERROR</td>
<td>such as format or syntax. This can occur in the following situations:</td>
</tr>
<tr>
<td></td>
<td>Server encounters a decoding error while parsing the incoming request.</td>
</tr>
<tr>
<td></td>
<td>The request is an add or modify request that specifies the addition of</td>
</tr>
<tr>
<td></td>
<td>an attribute type to an entry but no values specified. Error reading</td>
</tr>
<tr>
<td></td>
<td>SSL credentials. An unknown type of modify operation is specified</td>
</tr>
<tr>
<td></td>
<td>(other than LDAP_MOD_ADD, LDAP_MOD_DELETE, and LDAP_MOD_REPLACE)</td>
</tr>
<tr>
<td></td>
<td>Unknown search scope.</td>
</tr>
<tr>
<td>03: LDAP_TIMELIMIT_</td>
<td>Search took longer than the time limit specified. If you have not</td>
</tr>
<tr>
<td>EXCEEDED</td>
<td>specified a time limit for the search, Oracle Internet Directory</td>
</tr>
<tr>
<td></td>
<td>uses a default time limit of one hour.</td>
</tr>
<tr>
<td>04: LDAP_SIZELIMIT_</td>
<td>More entries match the search query than the size limit specified.</td>
</tr>
<tr>
<td>EXCEEDED</td>
<td>If you have not specified a size limit for the search, Oracle</td>
</tr>
<tr>
<td></td>
<td>Internet Directory uses a default size limit of 1000.</td>
</tr>
<tr>
<td>05: LDAP_COMPARE_</td>
<td>Presented value is not the same as the one in the entry.</td>
</tr>
<tr>
<td>FALSE</td>
<td></td>
</tr>
<tr>
<td>06: LDAP_COMPARE_</td>
<td>Presented value is same as the one in the entry.</td>
</tr>
<tr>
<td>TRUE</td>
<td></td>
</tr>
<tr>
<td>07: LDAP_STRONG_</td>
<td>The requested bind method is not supported by the server. For example,</td>
</tr>
<tr>
<td>AUTH_NOT_SUPPORTED</td>
<td>SASL clients requesting Kerberos authentication from Oracle Internet</td>
</tr>
<tr>
<td></td>
<td>Directory receive this error in response.</td>
</tr>
<tr>
<td>09: LDAP_PARTIAL_</td>
<td>Server returned a referral.</td>
</tr>
<tr>
<td>RESULTS</td>
<td></td>
</tr>
<tr>
<td>10: LDAP_REFERRAL</td>
<td>Server returned a referral.</td>
</tr>
<tr>
<td>12: LDAP_UNAVAILABLE_</td>
<td>Specified request is not supported</td>
</tr>
<tr>
<td>CRITICALEXTENSION</td>
<td></td>
</tr>
<tr>
<td>Error</td>
<td>Cause</td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>16: LDAP_NO_SUCH_ATTRIBUTE</td>
<td>Attribute does not exist in the entry specified in the request.</td>
</tr>
<tr>
<td>17: LDAP_UNDEFINED_TYPE</td>
<td>Specified attribute type is undefined in the schema.</td>
</tr>
<tr>
<td>19: LDAP_CONSTRAINT_VIOLATION</td>
<td>The value in the request violated certain constraints.</td>
</tr>
<tr>
<td>20: LDAP_TYPE_OR_VALUE_EXISTS</td>
<td>Duplicate values specified for the attribute.</td>
</tr>
<tr>
<td>21: LDAP_INVALID_SYNTAX</td>
<td>Specified attribute syntax is invalid. In a search, the filter syntax is invalid.</td>
</tr>
<tr>
<td>32: LDAP_NO_SUCH_OBJECT</td>
<td>The base specified for the operation does not exist.</td>
</tr>
<tr>
<td>34: LDAP_INVALID_DN_SYNTAX</td>
<td>Error in the DN syntax.</td>
</tr>
<tr>
<td>49: LDAP_INVALID_CREDENTIALS</td>
<td>Bind failed because the credentials are not correct.</td>
</tr>
<tr>
<td>50: LDAP_INSUFFICIENT_ACCESS</td>
<td>The client does not have access to perform this operation.</td>
</tr>
<tr>
<td>53: LDAP_UNWILLING_TO_PERFORM</td>
<td>General error, or server is in read-only mode.</td>
</tr>
<tr>
<td>65: LDAP_OBJECT_CLASS_VIOLATION</td>
<td>A change to the entry violates the object class definition.</td>
</tr>
<tr>
<td>66: LDAP_NOT_ALLOWED_ON_NONLEAF</td>
<td>The entry to be deleted has children.</td>
</tr>
<tr>
<td>67: LDAP_NOT_ALLOWED_ON_RDN</td>
<td>Cannot perform the operation on RDN attributes—for example, you cannot delete the RDN attribute of the entry.</td>
</tr>
<tr>
<td>68: LDAP_ALREADY_EXISTS</td>
<td>Duplicate ADD condition.</td>
</tr>
<tr>
<td>81: LDAP_SERVER_DOWN</td>
<td>Cannot contact the directory server. This message is returned from the SDK.</td>
</tr>
<tr>
<td>82: LDAP_LOCAL_ERROR</td>
<td>The client encountered an internal error. This message is returned from the client SDK.</td>
</tr>
<tr>
<td>83: LDAP_ENCODING_ERROR</td>
<td>The client encountered an error in encoding the request. This message is returned from the SDK.</td>
</tr>
<tr>
<td>84: LDAP_DECODING_ERROR</td>
<td>The client encountered an error in decoding the request. This message is returned from the SDK.</td>
</tr>
<tr>
<td>85: LDAP_TIMEOUT</td>
<td>Client encountered the time out specified for the operation. This message is returned from the SDK.</td>
</tr>
<tr>
<td>86: LDAP_AUTH_UNKNOWN</td>
<td>Authentication method is unknown to the client SDK.</td>
</tr>
<tr>
<td>87: LDAP_FILTER_ERROR</td>
<td>Bad search filter</td>
</tr>
<tr>
<td>88: LDAP_USER_CANCELLED</td>
<td>User cancelled operation</td>
</tr>
<tr>
<td>89: LDAP_PARAM_ERROR</td>
<td>Bad parameter to an LDAP routine</td>
</tr>
<tr>
<td>90: LDAP_NO_MEMORY</td>
<td>Out of memory</td>
</tr>
</tbody>
</table>
Additional Directory Server Error Messages

Table L–2 lists additional directory server error messages and their causes. These messages do not display error codes.

The Oracle Internet Directory application replaces the parameter tag seen in some of the following messages with the appropriate runtime value.

### Table L–2 Additional Error Messages

<table>
<thead>
<tr>
<th>Error</th>
<th>Cause</th>
</tr>
</thead>
<tbody>
<tr>
<td>%s attribute not found</td>
<td>The particular attribute type is not defined in the schema.</td>
</tr>
<tr>
<td>&lt;parameter&gt; not found for attribute &lt;parameter&gt;</td>
<td>Value not found in the attribute. (ldapmodify)</td>
</tr>
<tr>
<td>Admin domain does not contain schema information for objectclass &lt;parameter&gt;</td>
<td>The object class specified in the request is not present in the schema.</td>
</tr>
<tr>
<td>Attempted to add a Class with oid &lt;parameter&gt; taken by other class</td>
<td>Duplicate object identifier specified. (schema modification)</td>
</tr>
<tr>
<td>Attribute &lt;parameter&gt; already in use</td>
<td>Duplicate attribute name. (schema modification)</td>
</tr>
<tr>
<td>Attribute &lt;parameter&gt; has syntax error.</td>
<td>Syntax error in the attribute name definition. (schema modification)</td>
</tr>
<tr>
<td>Attribute &lt;parameter&gt; is not supported in the schema.</td>
<td>Attribute not defined. (all operations)</td>
</tr>
<tr>
<td>Attribute &lt;parameter&gt; is single valued.</td>
<td>Attribute is single-valued. (ldapadd and ldapmodify)</td>
</tr>
<tr>
<td>Attribute &lt;parameter&gt; not present in the entry.</td>
<td>This attribute does not exist in the entry. (ldapmodify)</td>
</tr>
<tr>
<td>Bad attribute definition.</td>
<td>Syntax error in attribute definition. (schema modification)</td>
</tr>
<tr>
<td>Currently Not Supported</td>
<td>The version of LDAP request is not supported by this server.</td>
</tr>
<tr>
<td>Entry to be deleted not found.</td>
<td>DN specified in the delete operation not found.</td>
</tr>
<tr>
<td>Entry to be modified not found</td>
<td>The entry specified in the request is not found.</td>
</tr>
<tr>
<td>Error encountered while adding &lt;parameter&gt; to the entry</td>
<td>Returned when modify add operation is invoked. A possible cause is that the system resource is unavailable.</td>
</tr>
<tr>
<td>Error encountered while encrypting an attribute value.</td>
<td>Error in encrypting user password. (all operations)</td>
</tr>
<tr>
<td>Error in DN Normalization.</td>
<td>DN specified is invalid. Syntax error encountered in parsing the DN. (all operations)</td>
</tr>
<tr>
<td>Error in hashing &lt;parameter&gt; attribute.</td>
<td>Error in creating hash entry for the attribute. (schema modification)</td>
</tr>
<tr>
<td>Error in hashing &lt;parameter&gt; objectclass.</td>
<td>Error in creating hash entry for the objectclass. (schema modification)</td>
</tr>
<tr>
<td>Error in Schema hash creation.</td>
<td>Error while creating hash table for schema. (schema modification)</td>
</tr>
<tr>
<td>Error</td>
<td>Cause</td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>Error replacing <code>&lt;parameter&gt;</code></td>
<td>Error in replacing this attribute. (ldapmodify)</td>
</tr>
<tr>
<td>Error while normalizing value for attribute <code>&lt;parameter&gt;</code></td>
<td>Error in normalizing value for the attribute. (all operations)</td>
</tr>
<tr>
<td>Failed to find <code>&lt;parameter&gt;</code> in mandatory or optional attribute list</td>
<td>Attribute specified does not exist in either the mandatory or optional attribute list as required by the object class(es).</td>
</tr>
<tr>
<td>Function Not Implemented</td>
<td>The feature/request is currently not supported.</td>
</tr>
<tr>
<td>INVALID ACI is <code>&lt;parameter&gt;</code></td>
<td>The particular ACI you specified in a request is invalid.</td>
</tr>
<tr>
<td>Mandatory attribute <code>&lt;parameter&gt;</code> is not defined in Admin Domain <code>&lt;parameter&gt;</code></td>
<td>MUST refers to attribute not defined. (schema modification)</td>
</tr>
<tr>
<td>Mandatory Attribute missing</td>
<td>The mandatory attribute for the particular entry is missing, as required by the particular object class.</td>
</tr>
<tr>
<td>Matching rule, <code>&lt;parameter&gt;</code>, not defined</td>
<td>Matching rule not defined in the server. (schema modification)</td>
</tr>
<tr>
<td>MaxConn Reached</td>
<td>The maximum number of concurrent connections to the LDAP server has been reached.</td>
</tr>
<tr>
<td>Modifying the Naming attribute for the entry without modifying the DN</td>
<td>Cannot modify the naming attributes using ldapmodify. A naming attribute, such as <code>cn</code> is an element in the DN.</td>
</tr>
<tr>
<td>New Parent not found</td>
<td>New parent specified in modifydn operation does not exist. (ldapmodifydn)</td>
</tr>
<tr>
<td>Object already exists</td>
<td>Duplicate entry. (ldapadd and ldapmodifydn)</td>
</tr>
<tr>
<td>Object ID <code>&lt;parameter&gt;</code> already in use</td>
<td>Duplicate object identifier specified. (schema modification)</td>
</tr>
<tr>
<td>Objectclass <code>&lt;parameter&gt;</code> already in use</td>
<td>Duplicate Objectclass name. (schema modification)</td>
</tr>
<tr>
<td>Objectclass attribute missing</td>
<td>The objectclass attribute is missing for this particular entry.</td>
</tr>
<tr>
<td>OID <code>&lt;parameter&gt;</code> has syntax error</td>
<td>Syntax error in the object identifier definition. (schema modification)</td>
</tr>
<tr>
<td>One of the attributes in the entry has duplicate value</td>
<td>You entered two values for the same attribute in the entry you are creating.</td>
</tr>
<tr>
<td>Operation not allowed on the <code>&lt;parameter&gt;</code></td>
<td>Operation not allowed on this entry. (modify, add, and delete)</td>
</tr>
<tr>
<td>Operation not allowed on the DSE Entry</td>
<td>Can’t do this operation on DSE entry. (delete)</td>
</tr>
<tr>
<td>Optional attribute <code>&lt;parameter&gt;</code> is not defined in Admin Domain <code>&lt;parameter&gt;</code></td>
<td>MAY refers to attribute not defined. (schema modification)</td>
</tr>
<tr>
<td>Parent entry not found in the directory</td>
<td>Parent entry does not exist. (ldapadd and perhaps ldapmodifydn)</td>
</tr>
</tbody>
</table>
Troubleshooting Password Policies

This section describes error messages and problems related to password policies.

Password Policy Error Messages

Table L–3 contains the error messages sent to the client as a result of password policy violations. The error codes are not standard LDAP error codes. They are messages sent as a part of additional information in the LDAP result.

<table>
<thead>
<tr>
<th>Error Number</th>
<th>Exception</th>
<th>Comment or Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>9000</td>
<td>GSL_PWDEXPIRED_EXCP</td>
<td>User's password has expired.</td>
</tr>
<tr>
<td>9001</td>
<td>GSL_ACCOUNTLOCKED_EXCP</td>
<td>User account is locked.</td>
</tr>
<tr>
<td>9002</td>
<td>GSL_EXPIREWARNING_EXCP</td>
<td>User password will expire in pwdexpirewarning seconds. Please change your password now.</td>
</tr>
<tr>
<td>9003</td>
<td>GSL_PWDMINLENGTH_EXCP</td>
<td>User password is not the required number of characters long.</td>
</tr>
<tr>
<td>9004</td>
<td>GSL_PWDNUMERIC_EXCP</td>
<td>User password does not contain required numeric characters.</td>
</tr>
<tr>
<td>9005</td>
<td>GSL_PWDNULL_EXCP</td>
<td>User password is a null password, which is disallowed.</td>
</tr>
<tr>
<td>9006</td>
<td>GSL_PWDINHISTORY_EXCP</td>
<td>User's new password is the same as an old one saved in history, which is disallowed. (The pwdinhistory attribute controls the number of passwords saved in history.)</td>
</tr>
<tr>
<td>9007</td>
<td>GSL_PWDILLEGALVALUE_EXCP</td>
<td>The user password supplied is an illegal value defined in orclpwdillegalvalues, and therefore cannot be used.</td>
</tr>
<tr>
<td>9008</td>
<td>GSL_GRACELOGIN_EXCP</td>
<td>User password has expired. User has pwdgraceloginlimit grace logins left or orclpwdgracelogintimelimit seconds in which grace logins are allowed.</td>
</tr>
<tr>
<td>9012</td>
<td>GSL_PWDALPHA_EXCP</td>
<td>Your Password must contain at least orclpwdalphanumeric alphabetic characters.</td>
</tr>
<tr>
<td>9013</td>
<td>GSL_PWDSPPECIAL_EXCP</td>
<td>Your Password must contain at least orclpwdspecialchars special characters.</td>
</tr>
</tbody>
</table>
Troubleshooting Directory Performance

This section gives some quick pointers for common performance-related problems.

Poor LDAP Search Performance

LDAP search performance is poor.

Problem

Various problems.

Solution

Make sure that:

- Schema associated with the ODS user is ANALYZED
- For searches involving multiple filter operands, make sure that the order in which they are given goes from the most specific to the least specific. For example, 
  
  & (uid=john.doe) (objectclass=person) is better than
  
  & (objectclass=person) (uid=john.doe).

See Also:  "Managing Password Policies, Accounts, and Passwords"  on page 19-8

Table L–3 (Cont.) Password Policy Violation Error Messages

<table>
<thead>
<tr>
<th>Error Number</th>
<th>Exception</th>
<th>Comment or Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>9014</td>
<td>GSL_PWDUPPER_EXCP</td>
<td>Your Password must contain at least orclpwdminuppercase upper case characters</td>
</tr>
<tr>
<td>9015</td>
<td>GSL_PWDMAXCHAR_EXCP</td>
<td>Your Password can only contain orclpwdmaxrptchars repeated characters</td>
</tr>
<tr>
<td>9016</td>
<td>GSL_PWDLOWER_EXCP</td>
<td>Your Password must contain at least orclpwdminlowercase lower case characters.</td>
</tr>
<tr>
<td>9017</td>
<td>GSL_EC_PWDPOLSUBENTRY_EXCP</td>
<td>The pwdPolicySubentry provided is invalid. (The DN is not that of a valid password policy present in the directory.)</td>
</tr>
<tr>
<td>9018</td>
<td>GSL_EC_PWDPOLINUSE</td>
<td>The pwdPolicy entry you are deleting is currently in use. (You must remove the references to the password policy before the policy itself can be removed.)</td>
</tr>
<tr>
<td>9019</td>
<td>GSL_EC_PWDPOLOBJ</td>
<td>The DN of a pwdPolicy entry may not be modified.</td>
</tr>
<tr>
<td>9020</td>
<td>GSL_PWDMINAGE_EXCP</td>
<td>Your Password has to be at least pwdminage seconds old before it can be changed.</td>
</tr>
<tr>
<td>9032</td>
<td>GSL_EC_GRACE_CONST</td>
<td>orclgracelogintimelimit and pwdgraceloginlimit are mutually exclusive. Both cannot be non-zero.</td>
</tr>
<tr>
<td>9033</td>
<td>GSL_EC_NOROOTDSEPPWDPOL</td>
<td>The pwdpoliciesubentry attribute in the root DSE cannot be deleted. (This is not allowed because it would leave the directory without an applicable password policy.)</td>
</tr>
<tr>
<td>9034</td>
<td>GSL_EC_NOTROCPWDPOL</td>
<td>Only password policies defined in the Root Oracle Context are applicable in the Root DSE. (This ensures that only a policy specified by an admin who has directory-wide privileges can be applied to the entire directory.)</td>
</tr>
<tr>
<td>9050</td>
<td>GSL_ACCTDISABLED_EXCP</td>
<td>User account has been disabled.</td>
</tr>
</tbody>
</table>
Problems and Solutions

Troubleshooting Oracle Internet Directory

L-9

Poor LDAP Add or Modify Performance
LDAP add or modify performance is poor.

Problem
Various problems

Solution
Make sure that:

■ There are enough redo log files in the database
■ The undo tablespace in the database is large enough
■ The schema associated with the ODS user is ANALYZED

When estimating the statistics, you can use the OID Database Statistics Collection tool to analyze the various database ODS schema objects.

Both the tracing functionality described in "Using Debug Logging" on page 14-2 and the database tracing event 10046 can assist you in diagnosing performance issues.

See Also: The oidstats.sql command-line tool reference in Oracle Identity Management User Reference for instructions on using the OID Database Statistics Collection tool

"Optimizing Searches" on page 25-9 for instructions on optimizing searches

MetaLink note 243006.1 on Oracle MetaLink, http://metalink.oracle.com, for information on performance issues with group entries

Troubleshooting Starting, Stopping, and Restarting of the Directory Server

To troubleshoot starting and stopping the directory server, you must know the purpose of each tool involved, how all the tools work together, and the overall process for starting and stopping the server.

See Also: Chapter 25, "Tuning Considerations for the Directory"

About the Tools for Starting, Stopping, and Restarting the Directory Server Instance

The tool used to start and stop the directory server as an Oracle Application Server component is OPMN. For information on troubleshooting OPMN, see the "Troubleshooting" appendix in Oracle Process Manager and Notification Server Administrator’s Guide.

There are two tools used to start, stop, and restart directory server instances: OID Control Utility (OIDCTL) and OID Monitor (OIDMON).

OIDCTL When OIDCTL is executed, it connects to the database as user ODS. Depending on the options used in the command, it either inserts or updates rows into a table named ODS.ODS_PROCESS. If the START option is used, then a row is inserted. If either the STOP or RESTART option is used, then a row is updated.

The ODS.ODS_PROCESS table includes the following information:

■ instance: The unique number of the instance, any value between 0 and 1000
■ pid: Process identifier, which will be updated by OIDMON when the process is started

Troubleshooting Oracle Internet Directory  L-9
- **state**: The type of operation requested
  
  The possible values for **state** are:
  - 0=stop
  - 1=start
  - 2=running
  - 3=restart
  - 4=shutdown
  - 5=failedover

**Note:** When OPMN is used to stop the directory server, the value for state is initially 4, that is, shutdown. However, once OPMN starts the directory server again, the state value becomes 2, that is, running.

**OIDMON** To start, stop, or restart a directory server instance, OIDMON must be running. At specified intervals, this daemon checks the value of the **state** column in the **ODS.ODS_PROCESS** table.

If it finds a row with **state=0**, then it reads the **pid** and stops the process.

If it finds one with **state=1** or **state=4**, then it starts a new process and updates the **pid** column with a new process identifier.

If it finds one with **state=2**, then it reads the **pid** and verifies that the process with that **pid** is running. If it is not running, then OIDMON starts a new process and updates the **pid** column with a new process identifier.

If it finds a row with **state=3**, then OIDMON reads the **pid**, stops the process, starts a new one, and updates the **pid** accordingly.

If OIDMON cannot start the server for some reason, it retries. If it is not running on a node in an Oracle Application Server Cluster (Identity Management) configuration, and it is still unsuccessful after 10 retries, it deletes the row from the **ODS.ODS_PROCESS** table. If OIDMON is running on a node in an Oracle Application Server Cluster (Identity Management) configuration, it retries 100 times.

If it is still unsuccessful, it pushes the request to another node.

In short, OIDCTL inserts and updates state information in the rows in the **ODS.ODS_PROCESS** table. OIDMON then reads that information and performs the specified task.

**About the Processes Involved in Starting, Stopping, and Restarting the Directory Server**

Starting, stopping and restarting the directory server involves a number of processes. OIDMON is one process. On Unix, it is called **oidmon**. In a Microsoft Windows environment, it is called **oidmon.exe**.

To start an instance, OIDMON checks the unique number in the **instance** column mentioned in the previous section. It then starts another process, namely, the listener/dispatcher, which is different from the Oracle Net Services listener process. It stores the process identifier for that new process in the **pid** column.
The listener/dispatcher, in turn, starts a number of server processes as defined in the configuration set entry. Note that these server processes are controlled by the listener/dispatcher and not by OIDMON. If one of these processes fails, then it is automatically restarted by the listener/dispatcher.

Together, the listener/dispatcher and the server processes constitute a directory server instance. On UNIX, this directory server instance is called oidldapd. On Microsoft Windows, they are called oidldapd.exe.

In short, there are at least three processes: one for OIDMON and at least two for the directory server itself. When all processes are running, you should see something like the following on UNIX computers:

```
% ps -ef|grep oid
root 12387 12381 0 Mar 28 0:05 oidldapd -i 1 -conf 0 key=811436710
root 12381 1 0 Mar 28 0:10 oidmon start
root 13297 1 0 Mar 28 0:14 oidldapd
```

Another way to obtain server information is by running ldapcheck. When you do this, you may see something like this:

```
Checking Oracle Internet Directory Processes ...
Process oidmon is Alive as PID 12381
Process oidldapd is Alive as PID 12387
Process oidldapd is Alive as PID 13297
Not Running ---- Process oidrepld
```

### Problems Starting, Stopping, and Restarting the Directory Server

This section describes some problems you might have when starting, stopping, or restarting the directory server.

**OIDCTL or OIDMON fails** Either OIDCTL or OIDMON can fail for a number of reasons.

**Problem** Incorrect syntax

**Solution** Verify that you are using the correct syntax as described in "Oracle Internet Directory Server Administration Tools" in Oracle Identity Management User Reference. Note that the correct value of the connect option when using OIDCTL is the TNS alias—that is, the connect string—and not a host name or other value. See Oracle MetaLink note 155790.1, on Oracle MetaLink, [http://metalink.oracle.com](http://metalink.oracle.com).

**Problem** The Oracle Internet Directory-designated database is not running.

**Solution** Verify that the Oracle Internet Directory-designated database and the Oracle Net Services components are correctly configured and running. To do this, see if you can connect to the database by using SQL*Plus that is installed in the same ORACLE_HOME as OIDCTL. Log in as ODS/ods_password@tns_alias where tns_alias is the same as that used in the connect option with OIDCTL. See Oracle MetaLink note 155790.1, on Oracle MetaLink, [http://metalink.oracle.com](http://metalink.oracle.com).
Problem
LDAP name resolution requires two instances of Oracle Internet Directory, but only one is running.

Solution
Verify that the value of the DIRECTORY_SERVERS parameter in the file ldap.ora is different from that specified in NAMES.DIRECTORY_PATH in the file sqlnet.ora. Both of these files are found in ORACLE_HOME/network/admin. If everything is working correctly, then selecting from ODS.ODS_PROCESS retrieves rows with state values described in “OIDCTL” on page L-9. See Oracle MetaLink, http://metalink.oracle.com.

Information in ODS.ODS_PROCESS is correct, but processes still do not start.
When everything is working correctly, you should see at least three processes: one named oidmon, and at least two named oidldapd. OIDMON starts, stops, and restarts the server processes, and, because it does so at specified intervals, give it time to complete the requested operation.

Problem
Missing oidldapd file.

Solution
See oidmon.log. Look for the message: No such file or directory. To correct the problem, replace the executable file.

Problem
Wrong permissions on oidldapd executable file.

Solution
Look for the message Exec of OIDLDAPD failed with error 13. On UNIX, the $ORACLE_HOME/bin/oidldapd file must have the following permissions:
-rws--x--- 1 root dba 1691802 Jan 20 10:30 oidldapd

If the permissions are not correct, type the following, as root:
cd $ORACLE_HOME/bin
chown root:dba oidldapd
chmod 0710 oidldapd
chmod u+s oidldapd

Problem
You are running as a user with insufficient privilege

Solution
To confirm that this is the problem, see oidmon.log. Look for the message: Permission denied or Open Wallet failed. This happens if you are not running either as root or as the user who is in the dba group. To correct the problem, try again as the correct user.

Problem
A port is in use.
Solution
See oidldapdXX.log, where XX is the server instance number. Look for the message:
Bind failed on... This indicates that the port that oidldapd is configured to 
listen on is in use by some other process. To determine which process is using the port, 
type:

```
netstat -a | grep portNum
```

If necessary, reconfigure the other process to use a different port or configure 
oidldapd to listen on another port by adding a configset. Remember that, by 
default, oidldapd listens on two ports, an SSL and non-SSL port.

Problem
On a cluster or Oracle Application Server Cluster (Identity Management) 
configuration, OIDMON pushes the server to another node in a cluster when it cannot 
start the server on the local node.

Solution
See oidmon.log. Look for the message: gslsgfrPushServer: Could not 
start server on NodeA, trying to start on nodeNodeB. To correct this 
problem, you must first determine why OIDMON cannot start the server on the local 
node.

Problem
A possible problem with Oracle Net Services or with the database itself.

Solution
See oidmon.log, oidsrv.log, oidldapdxx.log, where xx is the server instance 
number, and oidrepdxx.log where XX is Oracle directory integration and 
provisioning server instance number, for details about the problem.

A Row is Missing from ODS.ODS_PROCESS

Problem
In a cluster or Oracle Application Server Cluster (Identity Management) configuration, 
OIDMON successfully starts oidldapd on both nodes, but then initiates failover due 
to a time stamp difference.

Solution
See oidmon.log. On the node with the missing row, look for the message:
Successfully failed over from NodeA to NodeB. On the other node, you 
will see an extra oidldapd. To correct the problem, adjust the system time on all 
nodes so that they are all within 250 seconds of one another.

Solution
See the trace files oidldapdxx.log where xx is the instance number, and 
oidldapdxxsyy.log where xx is the instance number and yy is the process 
identifier. If the trace files do not give useful information or pointers to Oracle 
MetaLink documents, then do the following: (1) Stop the directory server processes; (2) 
Remove or rename old trace files; (3) Start OIDMON and a directory server with 
maximum debug level, namely, 11744051. Note that, to get the trace files, you must 
first stop, then start, the server; you cannot simply restart it. Investigate the new trace 
files, and, if needed, log an iTAR with Oracle Support Services and upload the trace


**OIDCTL Error**  No processes are running, but using OIDCTL gives an error saying that the specified instance is already in use

**Problem**  This can occur, for example, after a machine restart when OIDMON is not running.

**Solution**  Start OIDMON, which, in turn, starts the directory server. See Oracle MetaLink note 155790.1, on Oracle MetaLink, http://metalink.oracle.com.

**Solution**  Use the stop option of OIDCTL to stop the specified instance. See Oracle MetaLink note 155790.1, on Oracle MetaLink, http://metalink.oracle.com.

**Solution**  If the directory server fails to start, you can override all user-specified configuration parameters to start it and then return the configuration sets to a workable state by using the ldapmodify operation. Use command-line options to oidctl to start the server with different configuration values, overriding any defined configuration sets except for the values in configset0. Do not modify configset0 because this technique relies on its minimal, default contents.

**Solution**  To see debug log files generated by the OID Control Utility, navigate to $ORACLE_HOME/ldap/log.

See Also: The oidctl command-line tool reference in Oracle Identity Management User Reference for more information on failover.

**Troubleshooting Oracle Internet Directory Replication**

This section discusses directory replication problems.

Whenever you investigate a replication problem, be sure to consult the log files $ORACLE_HOME/ldap/oidrepld00.log and oidldapdxx.log for information.

The replication server supports multiple debugging levels. To turn on replication debugging, specify the -d decimal_debug_level flag when you start the server. For example:

oidctl server=oidrepld connect=connect_string instance=instance_number \
flags="-h host -p port -d decimal_debug_level"

**Note:**  Turning on debugging will affect replication performance.
Problems and Solutions

Troubleshooting Oracle Internet Directory

Replication Server Does Not Start
There are several problems that can prevent the replication server from starting.

Problem
Invalid oidctl syntax

Solution
Use the following syntax to start the replication server:

oidctl server=oidrepld connect=connect_string instance=instance_number \ flags="-h host -p port"

Problem
Oracle Internet Directory is not running at the host and port you specified on the command line when you attempted to start the replication server. This caused the anonymous bind to the target Oracle Internet Directory to fail.

Solution
Make sure the target Oracle Internet Directory is up and running at the specified host and port.

Problem
The replication server is attempting to bind to the host and port specified in either the orclreplicaprimaryurl or the orclreplicasecondaryurl attribute of the Replica entry, but Oracle Internet Directory is running at a different host or port.

Solution
If you decide to run Oracle Internet Directory at a different host or port, add the new information to the orclreplicasecondaryurl attribute of the replica entry, as follows:

1. Prepare a modification file, mod.ldif. For example, to change to host my.us.oracle.com and port 4444, you would specify:

   dn: orclreplicaid=replica_ID, cn=replication configuration
   changetype: modify
   add: orclreplicasecondaryurl
   orclreplicasecondaryurl: ldap://my.us.oracle.com:4444/

2. Run:

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

Problem
The ReplBind credential in the replication wallet $ORACLE_HOME/ldap/admin/oidrORACLE_SID is corrupt or invalid. That is, the password stored in the wallet is not the same as the password that is stored in the directory, or the wallet does not exist. This causes the replication bind to fail and the replication server to exit with an error.

You might see messages similar to this example in the file oidrepldXX.log:

See Also: Chapter 14, "Logging, Auditing, and Monitoring the Directory" for more information about debugging.

Solution

Use remtool to fix the replication bind credential in the replication wallet or to synchronize between Oracle Internet Directory and the replication wallet.

- `remtool -pchgpwd` changes the password of the replication DN of a replica. Use this option if you know the current replication DN password stored in the directory and you want to change it both in the directory and in the wallet.

- `remtool -presetpwd` resets the password or the replication DN of a replica. Use this option if you know the current replication DN password stored in the directory and you want to change it both in the directory and in the wallet.

- `remtool -pchgwalpwd` changes password of replication DN of a replica only in the wallet. Use this option if you know the replication DN password stored in the directory but you are not sure whether the wallet has the correct password or you want to create the wallet file.

All of these options will create a wallet if one does not already exist.

See Also:

- The `remtool` command-line tool reference in *Oracle Identity Management User Reference* for more information about using `remtool`

- The `oidpasswd` command-line tool reference in *Oracle Identity Management User Reference* for more information about using `oidpasswd`

Repository Creation Assistant Error

Problem

When you use the Oracle Application Server tool RepCA to load Oracle Internet Directory schema into an existing Oracle 10.1.0.3 Database, you might see the following error message in the `$ORACLE_HOME/assistants/repca/log/repca*log` file:

```
SP2-0332: Cannot create spool file.
```

Solution

This error message can be ignored.

Errors in Replication Bootstrap

A number of errors can occur in replication bootstrap.

Problem

Some of the naming contexts failed to be bootstrapped.
Solution

Identify the naming contexts that failed to be bootstrapped, and use the oidcmprec tool to reconcile them. Then resume replication by setting the consumer's replica state to ONLINE mode.

Problem

Various causes.

Solution

Identify the cause of the bootstrap failure and fix the cause, then restart bootstrapping by setting consumer's replica state to BOOTSTRAP mode.

Solution

To determine the exact cause of the error, examine the log file oidldapd.xx.log. Look for error messages like those in the following example:

```
2004/09/14:12:57:23 * Starting OIDREPLD against dlsun1418:4444...
2004/09/14:12:57:25 * Starting scheduler...
2004/09/14:12:57:26 * Start to BootStrap from supplier=dlsun1418_replica to consumer=dlsun1418_replica2
2004/09/14:12:57:27 * gslrbssSyncDIT:Replicating namingcontext=cn=oraclecontext......
2004/09/14:12:58:21 * gslrbssSyncDIT:Replicating namingcontext=cn=joe smith......
2004/09/14:12:58:23 * gslrbssSyncDIT:Sync failed for namingctx: cn=joe smith, only 1 entries retrieved
2004/09/14:12:58:23 * gslrbssSyncDIT:Replicating namingcontext=cn=oracleschemaversion......
2004/09/14:12:58:25 * gslrbssSyncDIT:Sync done successfully for namingctx: cn=oracleschemaversion, 10 entries matched
2004/09/14:12:58:51 * gslrbsbBootStrap: Failure occured when bootstrapping 1 out of 3 namingcontext(s) from the supplier
```

Identify the cause of the bootstrap failure and fix it. You can identify the naming contexts that caused the problem, then use oidcmprec to compare and reconcile the naming contexts. Once you have resolved the problem, start bootstrapping again by starting the Oracle Internet Directory replication server.

Problem

The Oracle Internet Directory server was shut down during the bootstrapping.

Solution

Make sure both the supplier Oracle Internet Directory and the consumer Oracle Internet Directory servers are up and running during replication bootstrapping.

Problem

Some of the entries being bootstrapped cannot be applied at the consumer due to a constraint violation.

Solution

Make sure the Oracle Internet Directory schema of the consumer are synchronized with those of the supplier before starting replication bootstrapping. When you add an
LDAP replica, remtool ensures that the Oracle Internet Directory schema on the consumer replica are synchronized with those on the supplier replica.

**Problem**
Improper replication filtering during bootstrapping. Replication supports excluding one or more attributes during bootstrapping. However, if a mandatory attribute of an entry is configured to be excluded, that entry cannot be applied at the consumer due to an objectclass violation.

**Solution**
Follow the replication naming context configuration rules in Chapter 30, "Oracle Internet Directory Replication Installation and Configuration" to configure replication filtering properly.

If you are debugging LDAP replication, you should become familiar with the LDAP replica states. If LDAP-based replication is configured, when the replication server starts, it reads the replica state from the local replica. The replication server behaves differently, depending upon the local replica state. LDAP replication errors appear in `oidldapdxx.log`

**See Also:** Appendix H, "LDAP Replica States"

**Problem**
When you restart the replication server after the replication server failed to bootstrap a naming context having more than 5000 entries, you may see error messages similar to this in the log file `oidrepld00.log`:

```
Error=DSA is unwilling to perform
2005/04/05:15:36:09 * gsrbssSyncDIT:Sync failed for namingctx: dc=com, only 0 entries retrieved
```

The replication server performs two steps during bootstrap operation. First, in the consumer, it deletes the naming contexts that it has to bootstrap. Second, it copies entries belonging to those naming contexts from supplier to consumer. Deletion by the replication server of a naming context having several thousands of entries results in a big transaction. The undo tablespace needs to have sufficient space to accommodate a big transaction. If the database's undo tablespace does not have sufficient space, it will result in an ORA-30036 error.

**Solution**
Either have the database administrator add more space to the undo tablespace, or use the `bulkdelete` tool to delete the required naming context before you start the replication server.

**Changes Are Not Replicated**
Changes are not replicated from one node to another.

**Problem**
The replication server has run out of table space

**Solution**
Look for the following message in the server log:
OCI Error ORA-1653 : ORA-01653: unable to extend table ODS.ASR_CHG_LOG by 8192 in tablespace OLTS_DEFAULT

Extend the table space and investigate why the table space keeps growing.

**Problem**
The target Oracle Internet Directory server is down.

**Solution**
Restart the target Oracle Internet Directory server.

**Problem**
Various causes

**Solution**
Make sure the replication server is started on all nodes, in multi-master replication, and at the consumer node in single-master or fan-out replication.

For multi-master Oracle Database Advanced Replication, use `remtool` to diagnostic and fix problems.

- `remtool -asrverify` verifies the correctness of a DRG setup and reports problems.
- `remtool -asrrectify` verifies the correctness of a DRG setup, reports problems, and attempts to rectify the problems.

Check the replication log and LDAP log for error messages and fix the cause of the error after investigation.

**See Also:** The `remtool` command-line tool reference in Oracle Identity Management User Reference for more information about using `remtool`.

**Replication Stops Working**

**Problem**
Data is not replicated between the replicas. In some cases, a working replication setup stops working after OID Human Intervention Queue entries are applied to one of the nodes. In other cases, adding or deleting a new replica causes problems or failures.

**Problem**
Various causes

**Solution**
See the following Oracle MetaLink notes on Oracle MetaLink, [http://metalink.oracle.com](http://metalink.oracle.com):

- Note 171693.1, "Resolving Conflicts"
- Note 122039.1, "Troubleshooting Basics for Advanced Replication"
- Note 213910.1, "Debugging OID Replication when ASR_CHG_LOG Never Gets Populated."

You can search for Oracle MetaLink notes by entering a term such as "replication" into the search box.
Troubleshooting SSL Setup

Problem
Setting up Oracle Internet Directory for one-way LDAP connections over SSL fails.

Solution
Do not set up the SSL port of configset 0 with wallet mode 2 or 3. If you do, you will break Oracle Delegated Administration Services and other services and applications that expect to communicate with Oracle Internet Directory on the encrypted SSL port.

To correctly configure and test Oracle Internet Directory for SSL, follow the instructions in Oracle Metalink note 178714.1, on Oracle MetaLink, http://metalink.oracle.com. Also see the SSL section of the tutorial "Getting Started with Oracle Internet Directory" at http://www.oracle.com/technology/obe/obe_as_10g.

This section discusses possible problems when configuring SSL.

Troubleshooting Change Log Garbage Collection

Both replication and Oracle Directory Integration Platform use change logs to propagate information from a supplier directory to a consumer directory. All change logs are stored in the table ods_chg_log. In addition, replication change logs are stored in asr_chg_log.

This section discusses possible problems you might encounter with change log garbage collection.

Change Logs Are Not Purged
Change logs grow very large.

Problem
Change logs are not being purged due to a replication issue. For example, if a replication server has been down for a few days, replication change logs will not be purged because they are needed for replication recovery.

Solution
Resolve the replication issue. See "Troubleshooting Oracle Internet Directory Replication" on page L-14".

Problem
The attribute orclpurgetargetage is set too high and there are one or more enabled but inactive change log subscribers that do not update orclLastAppliedChangeNumber in their subscriber profiles. Change number-based purging won’t purge change logs that are not yet consumed and time-based purging won’t purge them because they’re not old enough.

Solution
Set the attribute orclpurgetargetage to a smaller value so that change logs are purged sooner.
Solution
Disable inactive changelog subscribers so that change logs are purged by change log number-based purging. Locate such enabled but inactive subscriber profiles by examining the `orclLastAppliedChangeNumber` in all subscriber profiles by typing:

```
ldapsearch -v -p port -h host -D cn=orcladmin -w password \
   -b "cn=changelog subscriber,cn=oracle internet directory" \
   -s sub "objectclass=orclchangesubscriber" orcllastappliedchangenumber orclsubscriberdisable
```

Look for an entry that has `orclSubscriberDisabled` equal to zero and an `orclLastAppliedChangeNumber` value that never changes. If such an entry exists, and the change log garbage collector's `orclpurgetargetage` is zero or greater, delete the value of `orclpurgetargetage`. When `orclpurgetargetage` is not defined or less than zero, the garbage collector will purge changes applied by the replication server, even if another subscriber has not updated its `orclLastAppliedChangeNumber`.

**See Also:** [“Change Log Purging in Multimaster Replication” on page 26-5.](#)

Troubleshooting Dynamic Password Verifiers

Table L–4 lists and describes the error messages for dynamic password verifiers.

**Table L–4 Error Messages for Dynamic Password Verifiers**

<table>
<thead>
<tr>
<th>Error Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>9022</td>
<td>A reversible encrypted password is missing from the user entry.</td>
</tr>
<tr>
<td>9023</td>
<td>The crypto type specified in the LDAP request control is not supported.</td>
</tr>
<tr>
<td>9024</td>
<td>The username parameter is missing from the LDAP request control.</td>
</tr>
</tbody>
</table>

If the directory is able to compare verifiers, and the comparison evaluates as false, the directory sends the standard error `LDAP_COMPARE_FALSE` to the client. Similarly, if the user being authenticated lacks a directory entry, the directory sends the standard error `LDAP_NO_SUCH_OBJECT`.

**See Also:** [“Password Verifier Schema Elements” in Oracle Identity Management User Reference](#)

Troubleshooting Oracle Internet Directory Password Wallets

The Oracle Internet Directory Server has two password wallets: `oidpwdlldap1` and `oidpwdSID`.

The `oidpwdlldap1` file contains the DN and password of an ODS user in encrypted format. The Oracle Internet Directory server uses the credential to connect to the back end database at startup time.

**Oracle Internet Directory Server Does Not Start**

Either `oidctl` or `opmn` fails to start an Oracle Internet Directory server instance.
Problems and Solutions

Problem
The password stored in the oidpwdlldap1 wallet is not synchronized with the ODS password in the back end database.

Solution
Try to connect to the database again using the sqlplus command:

```
sqlplus ods /ods_password@connect_string
```

If the connection succeeds, try to synchronize the password in the wallet with the ODS password by using the oidpasswd tool to create a new wallet with the correct password. For example:

```
>> oidpasswd connect=connect_string create_wallet=true
```

If the connection attempt fails, you must login into the back end database as a database administrator and change the ODS password by using the sql command:

```
>> alter user ods identified by some_new_password
```

Then try to create a new oidpwdlldap1 to store the new password.

Solution
Try to start the Oracle Internet Directory server again.

The oidpwdrSID file contains the DN and password of a replica DN in an encrypted format. The Oracle Internet Directory replication server uses the credential to connect to the Oracle Internet Directory server at startup time.

This is an example of a replication password wallet, oidpwdrSID:

```
/------BEGIN REPL CREDENTIAL:cn=replication dn,orclreplicaid=qdinh-sun_adeldap,cn=replication configuration-----
ezNkZXMtY2JjLXBrY3M1cGFkfQUnaz0TsfzczP0nMLHcHAXchf5mJw+sb4y0bLvvw3RvSg7H
S7/WaKBJ02fSGRMfWAV+61lkRQ26g==
------END REPL CREDENTIAL:cn=replication dn,orclreplicaid=qdinh-sun_adeldap,cn=replication configuration-----/
```

Password Not Synchronized
Either oidctl or opmn fails to start an Oracle Internet Directory server instance and the replication server log file oidrepld00.log reports that it is not able to bind.

Problem
The replica DN password stored in the oidpwdrSID is not synchronized with the replica DN password in the Oracle Internet Directory server.

Solution
Try to connect to the Oracle Internet Directory server instance using the ldapbind command. Specify the replica DN stored in oidpwdrSID and the replica DN password. For example:

```
>> ldapbind -h host -p port -D "cn=replication dn,orclreplicaid=qdinh-sun_adeldap, cn=replication configuration" -w replica_dn_password
```

If the connection succeeds, then you can reset the password in the oidpwdrSID wallet using remtool with the option -pchgwalpwd, which changes the password of the replication DN of a replica only in the wallet. If you do not remember the replication
Problems and Solutions

T roubleshooting Oracle Internet Directory

L-23
dn password, then you can reset it using remtool with the option -prestpwd, which resets the password of the replication dn of a replica.

After resetting the replication password wallet, restart the replication server instance again a using opmnctl or oidctl.

Troubleshooting bulkload

Oracle highly recommends that you investigate and correct all errors thrown by bulkload before proceeding with the next step. If you ignore a bulkload error, you are likely to run into serious problems later.

Most bulkload errors occur during data load or during index creation.

Problem

The bulkload command-line tool fails during data load.

Solution

Restore the directory to the state it was in before the data load by using one of these methods:

- Use the bulkload recover option
- Restore the directory from a backup taken before you invoked bulkload.

Problem

The bulkload command-line tool fails during index creation.

Solution

Examine bulkload.log. Find and fix the specific issue that caused index creation failure. Run bulkload with the index option again.

Failure to correct index errors can cause duplicate entries or duplicate rows in the Oracle Internet Directory’s tables.

Problem

The bulkload command-line tool fails because of a broken connection to the database. This can occur, for example, due to a host crash or in to a failover in Real Application Clusters.

Solution

Follow the following procedure:

1. Ensure that the database is restarted properly.
2. If the bulkload invocation employed only the check="TRUE" or generate="TRUE" options, but not the load="TRUE" option, go to step 3.

   If it was the bulkload load="TRUE" option that failed, you must restore the database to its state before the failure. How you do that depends on whether or not you have a backup of the database before you issued the bulkload load="TRUE" command.

   - If you have a backup, use it to restore the database to its original state before you issued the bulkload command.
   - If you do not have a backup, use the bulkload recover command to return the database to its state before the bulkload load="TRUE" command.
3. Re-issue the `bulkload` command that failed.

**Troubleshooting bulkdelete and bulkmodify**

Oracle highly recommends that you investigate and correct all errors thrown by the bulk tools before proceeding with the next step.

**Problem**
The `bulkdelete` or `bulkmodify` command-line tool fails because of a broken connection to the database. This can occur, for example, due to a host crash or in to a failover in Real Application Clusters.

**Solution**
Ensure that the database is restarted properly. Then retry the `bulkdelete` or `bulkmodify` command that failed.

**Troubleshooting catalog**

Oracle highly recommends that you investigate and correct all errors thrown by the bulk tools before proceeding with the next step.

**Problem**
The `catalog` command-line tool fails because of a broken connection to the database. This can occur, for example, due to a host crash or in to a failover in Real Application Clusters.

**Solution**
Ensure that the database is restarted properly. Retry the `catalog` command that failed. If the original invocation employed the `add="TRUE"` option, the retry might fail because the first command partially completed. If the retry fails, use `catalog delete="TRUE"` to delete the attribute index, then retry the command again.

**Need More Help?**

You can find more solutions on Oracle MetaLink, [http://metalink.oracle.com](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:
3DES
See Triple Data Encryption Standard (3DES).

access control item (ACI)
Access control information represents the permissions that various entities or subjects have to perform operations on a given object in the directory. This information is stored in Oracle Internet Directory as user-modifiable operational attributes, each of which is called an access control item (ACI). An ACI determines user access rights to directory data. It contains a set of rules for controlling access to entries (structural access items) and attributes (content access items). Access to both structural and content access items may be granted to one or more users or groups.

access control list (ACL)
A list of resources and the usernames of people who are permitted access to those resources within a computer system. In Oracle Internet Directory, an ACL is a list of access control item (ACI) attribute values that is associated with directory objects. The attribute values on that list represent the permissions that various directory user entities (or subjects) have on a given object.

access control policy point (ACP)
A directory entry that contains access control policy information that applies downward to all entries at lower positions in the directory information tree (DIT). This information affects the entry itself and all entries below it. In Oracle Internet Directory, you can create ACPs to apply an access control policy throughout a subtree of your directory.

account lockout
A security feature that locks a user account if repeated failed logon attempts occur within a specified amount of time, based on security policy settings. Account lockout occurs in OracleAS Single Sign-On when a user submits an account and password combination from any number of workstations more times than is permitted by Oracle Internet Directory. The default lockout period is 24 hours.

ACI
See access control item (ACI).

ACL
See access control list (ACL).
ACP
See access control policy point (ACP).

administrative area
A subtree on a directory server whose entries are under the control of a single administrative authority. The designated administrator controls each entry in that administrative area, as well as the directory schema, access control list (ACL), and attributes for those entries.

Advanced Encryption Standard (AES)
Advanced Encryption Standard (AES) is a symmetric cryptography algorithm that is intended to replace Data Encryption Standard (DES). AES is a Federal Information Processing Standard (FIPS) for the encryption of commercial and government data.

advanced replication
See Oracle Database Advanced Replication.

advanced symmetric replication (ASR)
See Oracle Database Advanced Replication.

AES
See Advanced Encryption Standard (AES).

anonymous authentication
The process by which a 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 programming interface (API).

application programming interface (API)
A series of software routines and development tools that comprise an interface between a computer application and lower-level services and functions (such as the operating system, device drivers, and other software applications). APIs serve as building blocks for programmers putting together software applications. For example, LDAP-enabled clients access Oracle Internet Directory information through programmatic calls available in the LDAP API.

application service provider
Application Service Providers (ASPs) are third-party entities that manage and distribute software-based services and solutions to customers across a wide area network from a central data center. In essence, ASPs are a way for companies to outsource some or almost all aspects of their information technology needs.

artifact profile
An authentication mechanism which transmits data using a compact reference to an assertion, called an artifact, instead of sending the full assertion. This profile accommodates browsers which handle a limited number of characters.

ASN.1
Abstract Syntax Notation One (ASN.1) is an International Telecommunication Union (ITU) notation used to define the syntax of information data. ASN.1 is used to describe
structured information, typically information that is to be conveyed across some communications medium. It is widely used in the specification of Internet protocols.

**ASR**

See Oracle Database Advanced Replication.

**assertion**

An assertion is a statement used by providers in security domains to exchange information about a subject seeking access to a resource. Identity providers, as well as service providers, exchange assertions about identities to make authentication and authorization decisions, and to determine and enforce security policies protecting the resource.

**asymmetric algorithm**

A cryptographic algorithm that uses different keys for encryption and decryption. See also: public key cryptography.

**asymmetric cryptography**

See public key cryptography.

**attribute**

Directory attributes hold a specific data element such as a name, phone number, or job title. Each directory entry is comprised of 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**

Attribute types specify information about a data element, such as the data type, maximum length, and whether it is single-valued or multivalued. The attribute type provides the real-world meaning for a value, and specifies the rules for creating and storing specific pieces of data, such as a name or an e-mail address.

**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**

Attribute values are the actual data contained within an attribute for a particular entry. For example, for the attribute type email, an attribute value might be sally.jones@oracle.com.

**authentication**

The process of verifying the identity claimed by an entity based on its credentials. Authentication of a user is generally based on something the user knows or has (for example, a password or a certificate).

Authentication of an electronic message involves the use of some kind of system (such as public key cryptography) to ensure that a file or message which claims to originate
from a given individual or company actually does, and a check based on the contents of a message to ensure that it was not modified in transit.

**authentication level**
An OracleAS Single Sign-On parameter that enables you to specify a particular authentication behavior for an application. You can link this parameter with a specific authentication plugin.

**authentication plugin**
An implementation of a specific authentication method. OracleAS Single Sign-On has Java plugins for password authentication, digital certificates, Windows native authentication, and third-party access management.

**authorization**
The process of granting or denying access to a service or network resource. Most security systems are based on a two step process. The first stage is authentication, in which a user proves his or her identity. The second stage is authorization, in which a user is allowed to access various resources based on his or her identity and the defined authorization policy.

**authorization policy**
Authorization policy describes how access to a protected resource is governed. Policy maps identities and objects to collections of rights according to some system model. For example, a particular authorization policy might state that users can access a sales report only if they belong to the sales group.

**basic authentication**
An authentication protocol supported by most browsers in which a Web server authenticates an entity with an encoded user name and password passed via data transmissions. Basic authentication is sometimes called plaintext authentication because the base-64 encoding can be decoded by anyone with a freely available decoding utility. Note that encoding is not the same as encryption.

**Basic Encoding Rules (BER)**
Basic Encoding Rules (BER) are the standard rules for encoding data units set forth in ASN.1. BER is sometimes incorrectly paired with ASN.1, which applies only to the abstract syntax description language, not the encoding technique.

**BER**
See Basic Encoding Rules (BER).

**binding**
In networking, binding is the establishment of a logical connection between communicating entities.

In the case of Oracle Internet Directory, binding refers to the process of authenticating to the directory.

The formal set of rules for carrying a SOAP message within or on top of another protocol (underlying protocol) for the purpose of exchange is also called a binding.

**block cipher**
Block ciphers are a type of symmetric algorithm. A block cipher encrypts a message by breaking it down into fixed-size blocks (often 64 bits) and encrypting each block with a key. Some well known block ciphers include Blowfish, DES, and AES.
See also: stream cipher.

**Blowfish**

Blowfish is a symmetric cryptography algorithm developed by Bruce Schneier in 1993 as a faster replacement for DES. It is a block cipher using 64-bit blocks and keys of up to 448 bits.

**CA**

See Certificate Authority (CA).

**CA certificate**

A Certificate Authority (CA) signs all certificates that it issues with its private key. The corresponding Certificate Authority’s public key is itself contained within a certificate, called a CA Certificate (also referred to as a root certificate). A browser must contain the CA Certificate in its list of trusted root certificates in order to trust messages signed by the CA’s private key.

**cache**

Generally refers to an amount of quickly accessible memory in your computer. However, on the Web it more commonly refers to where the browser stores downloaded files and graphics on the user’s computer.

**CBC**

See cipher block chaining (CBC).

**central directory**

In an Oracle Directory Integration Platform environment, the directory that acts as the central repository. In an Oracle Directory Integration Platform environment, Oracle Internet Directory is the central directory.

**certificate**

A certificate is a specially formatted data structure that associates a public key with the identity of its owner. A certificate is issued by a Certificate Authority (CA). It contains the name, serial number, expiration dates, and public key of a particular entity. The certificate is digitally signed by the issuing CA so that a recipient can verify that the certificate is real. Most digital certificates conform to the X.509 standard.

**Certificate Authority (CA)**

A Certificate Authority (CA) is a trusted third party that issues, renews, and revokes digital certificates. The CA essentially vouches for an entity's identity, and may delegate the verification of an applicant to a Registration Authority (RA). Some well known Certificate Authorities (CAs) include Digital Signature Trust, Thawte, and VeriSign.

**certificate chain**

An ordered list of certificates containing one or more pairs of a user certificate and its associated CA certificate.

**certificate management protocol (CMP)**

Certificate Management Protocol (CMP) handles all relevant aspects of certificate creation and management. CMP supports interactions between public key infrastructure (PKI) components, such as the Certificate Authority (CA), Registration Authority (RA), and the user or application that is issued a certificate.
**certificate request message format (CRMF)**
Certificate Request Message Format (CRMF) is a format used for messages related to the life-cycle management of X.509 certificates, as described in the RFC 2511 specification.

**certificate revocation list (CRL)**
A Certificate Revocation List (CRL) is a list of digital certificates which have been revoked by the Certificate Authority (CA) that issued them.

**change logs**
A database that records changes made to a directory server.

**cipher**
See cryptographic algorithm.

**cipher block chaining (CBC)**
Cipher block chaining (CBC) is a mode of operation for a block cipher. CBC uses what is known as an initialization vector (IV) of a certain length. One of its key characteristics is that it uses a chaining mechanism that causes the decryption of a block of ciphertext to depend on all the preceding ciphertext blocks. As a result, the entire validity of all preceding blocks is contained in the immediately previous ciphertext block.

**cipher suite**
In Secure Sockets Layer (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.

**ciphertext**
Ciphertext is the result of applying a cryptographic algorithm to readable data (plaintext) in order to render the data unreadable by all entities except those in possession of the appropriate key.

**circle of trust**
A trust relationship among a set of identity providers and service providers that allows a principal to use a single federated identity and single sign-on (SSO) when conducting business transactions with providers within that set.

Businesses federate or affiliate together into circles of trust based on Liberty-enabled technology and on operational agreements that define trust relationships between the businesses.

See also: federated identity management (FIM), Liberty Alliance.

**claim**
A claim is a declaration made by an entity (for example, a name, identity, key, group, and so on).

**client SSL certificates**
A type of certificate used to identify a client machine to a server through Secure Sockets Layer (SSL) (client authentication).
cluster
A collection of interconnected usable whole computers that is used as a single computing resource. Hardware clusters provide high availability and scalability.

CMP
See certificate management protocol (CMP).

CMS
See Cryptographic Message Syntax (CMS).

code signing certificates
A type of certificate used to identify the entity who signed a Java program, Java Script, or other signed file.

cold backup
In Oracle Internet Directory, this refers to the procedure of adding a new directory system agent (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 Oracle Internet 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.

confidentiality
In cryptography, confidentiality (also known as privacy) is the ability to prevent unauthorized entities from reading data. This is typically achieved through encryption.

configset
See configuration set entry.

configuration set entry
An Oracle Internet 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 directory-specific entry (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 distinguished name (DN) of the root of a naming context.

**CRL**
See certificate revocation list (CRL).

**CRMF**
See certificate request message format (CRMF).

**cryptographic algorithm**
A cryptographic algorithm is a defined sequence of processes to convert readable data (plaintext) to unreadable data (ciphertext) and vice versa. These conversions require some secret knowledge, normally contained in a key. Examples of cryptographic algorithms include DES, AES, Blowfish, and RSA.

**Cryptographic Message Syntax (CMS)**
Cryptographic Message Syntax (CMS) is a syntax defined in RFC 3369 for signing, digesting, authenticating, and encrypting digital messages.

**cryptography**
The process of protecting information by transforming it into an unreadable format. The information is encrypted using a key, which makes the data unreadable, and is then decrypted later when the information needs to be used again. See also public key cryptography and symmetric cryptography.

**dads.conf**
A configuration file for Oracle HTTP Server that is used to configure a database access descriptor (DAD).

**DAS**
See Oracle Delegated Administration Services. (DAS).

**Data Encryption Standard (DES)**
Data Encryption Standard (DES) is a widely used symmetric cryptography algorithm developed in 1974 by IBM. It applies a 56-bit key to each 64-bit block of data. DES and 3DES are typically used as encryption algorithms by S/MIME.
**data integrity**
The guarantee that the contents of the message received were not altered from the contents of the original message sent.

See also: *integrity*.

**database access descriptor (DAD)**
Database connection information for a particular Oracle Application Server component, such as the OracleAS Single Sign-On schema.

**decryption**
The process of converting the contents of an encrypted message (ciphertext) back into its original readable format (plaintext).

**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 directory information tree (DIT).

**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 realm location**
An attribute in the root Oracle Context that identifies the root of the default identity management realm.

**defederation**
The act of unlinking a user's account from an identity provider or service provider.

**Delegated Administration Services**
See Oracle Delegated Administration Services.

**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.

**DER**
See Distinguished Encoding Rules (DER).

**DES**
See Data Encryption Standard (DES).

**DIB**
See directory information base (DIB).
Diffie-Hellman
Diffie-Hellman (DH) is a public key cryptography protocol that allows two parties to establish a shared secret over an unsecure communications channel. First published in 1976, it was the first workable public key cryptographic system.

See also: symmetric algorithm.

digest
See message digest.

digital certificate
See certificate.

digital signature
A digital signature is the result of a two-step process applied to a given block of data. First, a hash function is applied to the data to obtain a result. Second, that result is encrypted using the signer's private key. Digital signatures can be used to ensure integrity, message authentication, and non-repudiation of data. Examples of digital signature algorithms include DSA, RSA, and ECDSA.

Digital Signature Algorithm (DSA)
The Digital Signature Algorithm (DSA) is an asymmetric algorithm that is used as part of the Digital Signature Standard (DSS). It cannot be used for encryption, only for digital signatures. The algorithm produces a pair of large numbers that enable the authentication of the signatory, and consequently, the integrity of the data attached. DSA is used both in generating and verifying digital signatures.

See also: Elliptic Curve Digital Signature Algorithm (ECDSA).

directory
See Oracle Internet Directory, Lightweight Directory Access Protocol (LDAP), and X.500.

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 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 integration profile
In an Oracle Directory Integration Platform environment, an entry in Oracle Internet Directory that describes how Oracle Directory Integration Platform communicates with external systems and what is communicated.

Directory Manager
See Oracle Directory Manager.

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 Platform 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 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.

**directory-specific entry (DSE)**
An entry specific to a directory server. Different directory servers may hold the same **directory information tree (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 user agent (DUA)**
The software that accesses a directory service on behalf of the directory user. The directory user may be a person or another software element.

**DIS**
See **directory integration server**.

**Distinguished Encoding Rules (DER)**
Distinguished Encoding Rules (DER) are a set of rules for encoding ASN.1 objects in byte-sequences. DER is a special case of **Basic Encoding Rules (BER)**.

**distinguished name (DN)**
A X.500 distinguished name (DN) is a unique name for a node in a directory tree. A DN is used to provide a unique name for a person or any other directory entry. A DN is a concatenation of selected **attributes** from each node in the tree along the path from the root node to the named entry’s node. For example, in LDAP notation, the DN for a person named John Smith working at Oracle's US office would be: "cn=John Smith, ou=People, o=Oracle, c=us".

**DIT**
See **directory information tree (DIT)**.

**DN**
See **distinguished name (DN)**.
**Document Type Definition (DTD)**
A Document Type Definition (DTD) is a document that specifies constraints on the tags and tag sequences that are valid for a given XML document. DTDs follow the rules of Simple Generalized Markup Language (SGML), the parent language of XML.

**domain**
A domain includes the Web site and applications that enable a principal to utilize resources. A federated site acts as an identity provider (also known as the source domain), a service provider (also known as the destination domain), or both.

**domain component attribute**
The domain component (dc) attribute can be used in constructing a distinguished name (DN) from a domain name. For example, using a domain name such as "oracle.com", one could construct a DN beginning with "dc=oracle, dc=com", and then use this DN as the root of its subtree of directory information.

**DRG**
See directory replication group (DRG).

**DSA**
See Digital Signature Algorithm (DSA) or directory system agent (DSA).

**DSE**
See directory-specific entry (DSE).

**DTD**
See Document Type Definition (DTD).

**ECC**
See Elliptic Curve Cryptography (ECC).

**ECDSA**
See Elliptic Curve Digital Signature Algorithm (ECDSA).

**EJB**
See Enterprise Java Bean (EJB).

**Elliptic Curve Cryptography (ECC)**
Elliptic Curve Cryptography (ECC) is an alternative to the RSA encryption system which is based on the difficulty of solving elliptic curve discrete logarithm problems rather than on factoring large numbers. Developed and marketed by Certicom, ECC is especially suitable for environments, such as wireless devices and PC cards, where computational power is limited and high speed is required. For any given key size (measured in bits) ECC provides more security (is harder to decrypt without the key) than RSA.

**Elliptic Curve Digital Signature Algorithm (ECDSA)**
The Elliptic Curve Digital Signature Algorithm (ECDSA) is the elliptic curve analog of the Digital Signature Algorithm (DSA) standard. The advantages of ECDSA compared to RSA-like schemes are shorter key lengths and faster signing and decryption. For example, a 160 (210) bit ECC key is expected to give the same security as a 1024 (2048) bit RSA key, and the advantage increases as level of security is raised.
**encryption**

Encryption is the process of converting plaintext to ciphertext by applying a **cryptographic algorithm**.

**encryption certificate**

An encryption certificate is a **certificate** containing a **public key** that is used to encrypt electronic messages, files, documents, or data transmission, or to establish or exchange a session key for these same purposes.

**end-to-end security**

This is a property of message-level security that is established when a message traverses multiple applications within and between business entities and is secure over its full route through and between the business entities.

**Enterprise Java Bean (EJB)**

Enterprise JavaBeans (EJBs) are a Java API developed by Sun Microsystems that defines a component architecture for multi-tier client/server systems. Because EJB systems are written in Java, they are platform independent. Being object oriented, they can be implemented into existing systems with little or no recompiling and configuring.

**Enterprise Manager**

See **Oracle Enterprise Manager**.

**entry**

An entry is a unique record in a directory that describes an object, such as a person. An entry consists of **attributes** and their associated **attribute values**, as dictated by the **object class** that describes that entry object. All entries in an LDAP directory structure are uniquely identified through their **distinguished name (DN)**.

**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 Platform server. Oracle Directory Integration Platform 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 Oracle Directory Integration Platform.

**external application**

Applications that do not delegate authentication to the OracleAS Single Sign-On server. Instead, they display HTML login forms that ask for application user names and passwords. At the first login, users can choose to have the OracleAS Single
Sign-On server retrieve these credentials for them. Thereafter, they are logged in to these applications transparently.

**failover**
The process of failure recognition and recovery. In an Oracle Application Server Cold Failover Cluster (Identity Management), 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.

**Federal Information Processing Standards (FIPS)**
Federal Information Processing Standards (FIPS) are standards for information processing issued by the US government Department of Commerce's National Institute of Standards and Technology (NIST).

**federated identity management (FIM)**
The agreements, standards, and technologies that make identity and entitlements portable across autonomous domains. FIM makes it possible for an authenticated user to be recognized and take part in personalized services across multiple domains. It avoids pitfalls of centralized storage of personal information, while allowing users to link identity information between different accounts. Federated identity requires two key components: trust and standards. The trust model of federated identity management is based on circle of trust. The standards are defined by the Liberty Alliance Project.

**federation**
See identity federation.

**filter**
A filter is an expression that defines the entries to be returned from a request or search on a directory. Filters are typically expressed as DNs, for example: cn=susie smith,o=acme,c=us.

**FIM**
See federated identity management (FIM).

**FIPS**
See Federal Information Processing Standards (FIPS).

**forced authentication**
The act of forcing a user to reauthenticate if he or she has been idle for a preconfigured amount of time. Oracle Application Server Single Sign-On enables you to specify a global user inactivity timeout. This feature is intended for installations that have sensitive applications.

**GET**
An authentication method whereby login credentials are submitted as part of the login URL.
**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.

**global user inactivity timeout**
An optional feature of Oracle Application Server Single Sign-On that forces users to reauthenticate if they have been idle for a preconfigured amount of time. The global user inactivity timeout is much shorter than the single sign-out session timeout.

**globalization support**
Multilanguage support for graphical user interfaces. Oracle Application Server Single Sign-On supports 29 languages.

**globally unique user ID**
A numeric string that uniquely identifies a user. A person may change or add user names, passwords, and distinguished names, but her globally unique user ID always remains the same.

**grace login**
A login occurring within the specified period before password expiration.

**group search base**
In the Oracle Internet Directory default directory information tree (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.

See also: hash function.

**hash function**
In cryptography, a hash function or one-way hash function is an algorithm that produces a given value when applied to a given block of data. The result of a hash function can be used to ensure the integrity of a given block of data. For a hash
function to be considered secure, it must be very difficult, given a known data block and a known result, to produce another data block that produces the same result.

**Hashed Message Authentication Code (HMAC)**
Hashed Message Authentication Code (HMAC) is a hash function technique used to create a secret hash function output. This strengthens existing hash functions such as MD5 and SHA. It is used in transport layer security (TLS).

**HMAC**
See **Hashed Message Authentication Code (HMAC)**.

**HTTP**
The Hyper Text Transfer Protocol (HTTP) is the protocol used between a Web browser and a server to request a document and transfer its contents. The specification is maintained and developed by the World Wide Web Consortium.

**HTTP Redirect Profile**
A **federation** profile which indicates that the requested resource resides under a different URL.

**HTTP Server**
See **Oracle HTTP Server**.

**httpd.conf**
The file used to configure **Oracle HTTP Server**.

**iASAdmins**
The administrative group responsible for user and group management functions in Oracle Application Server. The OracleAS Single Sign-On administrator is a member of the group iASAdmins.

**identity federation**
The linking of two or more accounts a **principal** may hold with one or more identity providers or service providers within a given **circle of trust**.

When users federate the otherwise isolated accounts they have with businesses, known as their local identities, they create a relationship between two entities, an association comprising any number of service providers and identity providers.

See also: **identity provider**, **service provider**.

**identity management**
The process by which the complete security lifecycle 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 infrastructure database**
The database that contains data for OracleAS Single Sign-On and Oracle Internet Directory.
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.

identity provider
One of the three primary roles defined in the identity federation protocols supported by OSFS. The other primary roles are service provider and principal. The identity provider is responsible for managing and authenticating a set of identities within a given circle of trust.

A service provider, in turn, provides services or goods to a principal based on the identity provider’s authentication of a principal’s identity.

Identity providers are service providers offering business incentives so that other service providers affiliate with them. An identity provider typically authenticates and asserts a principal’s identity.

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.

infrastructure tier
The Oracle Application Server components responsible for identity management. These components are OracleAS Single Sign-On, Oracle Delegated Administration Services, and Oracle Internet Directory.

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**
In cryptography, integrity is the ability to detect if data has been modified by entities that are not authorized to modify it.

**Internet Directory**
See Oracle Internet Directory.

**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.

**J2EE**
See Java 2 Platform, Enterprise Edition (J2EE).

**Java 2 Platform, Enterprise Edition (J2EE)**
Java 2 Platform, Enterprise Edition (J2EE) is an environment for developing and deploying enterprise applications, defined by Sun Microsystems Inc. The J2EE platform consists of a set of services, application programming interfaces (APIs), and protocols that provide the functionality for developing multitiered, Web-based applications.

**Java Server Page (JSP)**
JavaServer Pages (JSP), a server-side technology, are an extension to the Java servlet technology that was developed by Sun Microsystems. JSPs have dynamic scripting capability that works in tandem with HTML code, separating the page logic from the static elements (the design and display of the page). Embedded in the HTML page, the Java source code and its extensions help make the HTML more functional, being used in dynamic database queries, for example.

**JSP**
See Java Server Page (JSP).

**key**
A key is a data structure that contains some secret knowledge necessary to successfully encrypt or decrypt a given block of data. The larger the key, the harder it is to crack a block of encrypted data. For example, a 256-bit key is more secure than a 128-bit key.

**key pair**
A public key and its associated private key.
See also: public/private key pair.

**knowledge reference**
The access information (name and address) for a remote directory system agent (DSA) and the name of the directory information tree (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

LDAP connection cache
To improve throughput, the OracleAS Single Sign-On server caches and then reuses connections to Oracle Internet Directory.

LDAP Data Interchange Format (LDIF)
A common, text-based format for exchanging directory data between systems. The set of standards for formatting an input file for any of the LDAP command-line utilities.

LDIF
See LDAP Data Interchange Format (LDIF).

legacy application
Older application that cannot be modified to delegate authentication to the OracleAS Single Sign-On server. Also known as an external application.

Liberty Alliance
The Liberty Alliance Project is an alliance of more than 150 companies, non-profit, and government organizations from around the globe. The consortium is committed to developing an open standard for federated network identity that supports all current and emerging network devices. The Liberty Alliance is the only global body working to define and drive open technology standards, privacy, and business guidelines for federated identity management (FIM).

Liberty ID-FF
Liberty Identity Federation Framework (Liberty ID-FF) provides an architecture for Web-based single sign-on (SSO) with federated identities.

Lightweight Directory Access Protocol (LDAP)
A set of protocols for accessing information in directories. LDAP supports TCP/IP, which is necessary for any type of Internet access. Its framework of design conventions supports industry-standard directory products, such as Oracle Internet Directory. Because it is a simpler version of the X.500 standard, LDAP is sometimes called X.500 light.

load balancer
Hardware devices and software that balance connection requests between two or more servers, either due to heavy load or failover. BigIP, Alteon, or Local Director are all popular hardware devices. Oracle Application Server Web Cache is an example of load balancing software.

logical host
In an Oracle Application Server Cold Failover Cluster (Identity Management), 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.
MAC
See message authentication code (MAC).

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 (MDS) 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.

MD2
Message Digest Two (MD2) is a message digest hash function. The algorithm processes input text and creates a 128-bit message digest which is unique to the message and can be used to verify data integrity. MD2 was developed by Ron Rivest for RSA Security and is intended to be used in systems with limited memory, such as smart cards.

MD4
Message Digest Four (MD4) is similar to MD2 but designed specifically for fast processing in software.

MD5
Message Digest Five (MD5) is a message digest hash function. The algorithm processes input text and creates a 128-bit message digest which is unique to the message and can be used to verify data integrity. MD5 was developed by Ron Rivest after potential weaknesses were reported in MD4. MD5 is similar to MD4 but slower because more manipulation is made to the original data.

MDS
See master definition site (MDS).

message authentication
The process of verifying that a particular message came from a particular entity.
See also: authentication.

**message authentication code (MAC)**

The Message Authentication Code (MAC) is a result of a two-step process applied to a given block of data. First, the result of a hash function is obtained. Second, that result is encrypted using a secret key. The MAC can be used to authenticate the source of a given block of data.

**message digest**

The result of a hash function. See also: hash.

**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.

**middle tier**

That portion of a OracleAS Single Sign-On instance that consists of the Oracle HTTP Server and OC4J. The OracleAS Single Sign-On middle tier is situated between the identity management infrastructure database and the client.

**mod_osso**

A module on the Oracle HTTP Server that enables applications protected by OracleAS Single Sign-On to accept HTTP headers in lieu of a user name and password once the user has logged into the OracleAS Single Sign-On server. The values for these headers are stored in the mod_osso cookie.

**mod_osso cookie**

User data stored on the HTTP server. The cookie is created when a user authenticates. When the same user requests another application, the Web server uses the information in the mod_osso cookie to log the user in to the application. This feature speeds server response time.

**mod_proxy**

A module on the Oracle HTTP Server that makes it possible to use mod_osso to enable single sign-on to legacy, or external applications.

**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.

**name identifier profile**

A federation profile which allows a provider to inform it’s peers when assigning or updating a name identifier for one of their common users.
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 references (also called referrals) to subordinate naming contexts. It can range in size from a single entry to the entire directory information tree (DIT).

native agent
In an Oracle Directory Integration Platform environment, an agent that runs under the control of the directory integration 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, for example:

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

Net Services
See Oracle Net Services.

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.

non-repudiation
In cryptography, the ability to prove that a given digital signature was produced with a given entity’s private key, and that a message was sent untampered at a given point in time.

OASIS
Organization for the Advancement of Structured Information Standards. OASIS is a worldwide not-for-profit consortium that drives the development, convergence and adoption of e-business standards.

object class
In LDAP, object classes are used to group information. Typically an object class models a real-world object such as a person or a server. Each directory entry belongs to one or
more object classes. The object class determines the attributes that make up an entry. One object class can be derived from another, thereby inheriting some of the characteristics of the other class.

**OC4J**

See Oracle Containers for J2EE (OC4).

**OCA**

See Oracle Certificate Authority.

**OCI**

See Oracle Call Interface (OCI).

**OCSP**

See Online Certificate Status Protocol (OCSP).

**OEM**

See Oracle Enterprise Manager.

**OID**

See Oracle Internet Directory.

**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 Internet Directory Server processes. It also controls the replication server if one is installed, and Oracle Directory Integration Platform Server.

**Online Certificate Status Protocol (OCSP)**

Online Certificate Status Protocol (OCSP) is one of two common schemes for checking the validity of digital certificates. The other, older method, which OCSP has superseded in some scenarios, is certificate revocation list (CRL). OCSP is specified in RFC 2560.

**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. See also: hash function.

**Oracle Application Server Single Sign-On**

OracleAS Single Sign-On consists of program logic that enables you to log in securely to applications such as expense reports, mail, and benefits. These applications take two
forms: **partner applications** and **external applications**. In both cases, you gain access to several applications by authenticating only once.

**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 Certificate Authority**

Oracle Application Server Certificate Authority is a **Certificate Authority (CA)** for use within your Oracle Application Server environment. OracleAS Certificate Authority uses Oracle Internet Directory as the storage repository for certificates. OracleAS Certificate Authority integration with OracleAS Single Sign-On and Oracle Internet Directory provides seamless certificate provisioning mechanisms for applications relying on them. A user provisioned in Oracle Internet Directory and authenticated in OracleAS Single Sign-On can choose to request a digital certificate from OracleAS Certificate Authority.

**Oracle CMS**

Oracle CMS implements the IETF **Cryptographic Message Syntax (CMS)** protocol. CMS defines data protection schemes that allow for secure message envelopes.

**Oracle Containers for J2EE (OC4J)**

A lightweight, scalable container for **Java 2 Platform, Enterprise Edition (J2EE)**.

**Oracle Context**

See [identity management realm-specific Oracle Context](#) and [root Oracle Context](#).

**Oracle Crypto**

Oracle Crypto is a pure Java library that provides core cryptography algorithms.

**Oracle Database Advanced Replication**

A feature in the Oracle Database that enables database tables to be kept synchronized across two Oracle databases.

**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 collection of interfaces and services for integrating multiple directories by using Oracle Internet Directory and several associated plug-ins and connectors. A feature of Oracle Internet Directory that enables an enterprise to use an external user repository to authenticate to Oracle products.

**Oracle Directory Integration 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 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 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 HTTP Server**
Software that processes Web transactions that use the Hypertext Transfer Protocol (HTTP). Oracle uses HTTP software developed by the Apache Group.

**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 Liberty SDK**
Oracle Liberty SDK implements the Liberty Alliance Project specifications enabling federated single sign-on between third-party Liberty-compliant applications.

**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 PKI SDK**
Oracle PKI SDK implements the security protocols that are necessary within public key infrastructure (PKI) implementations.

**Oracle SAML**
Oracle SAML provides a framework for the exchange of security credentials among disparate systems and applications in an XML-based format as outlined in the OASIS specification for the Security Assertions Markup Language (SAML).
**Oracle Security Engine**

**Oracle S/MIME**
Oracle S/MIME implements the Secure/Multipurpose Internet Mail Extension (S/MIME) specifications from the Internet Engineering Task Force (IETF) for secure e-mail.

**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 Web Services Security**
Oracle Web Services Security provides a framework for authentication and authorization using existing security technologies as outlined in the OASIS specification for Web Services Security.

**Oracle XML Security**
Oracle XML Security implements the W3C specifications for XML Encryption and XML Signature.

**OracleAS Portal**
An OracleAS Single Sign-On partner application that provides a mechanism for integrating files, images, applications, and Web sites. The External Applications portlet provides access to external applications.

**other information repository**
In an Oracle Directory Integration Platform environment, in which Oracle Internet Directory serves as the central directory, any information repository except Oracle Internet Directory.

**OWM**
See Oracle Wallet Manager.

**partition**
A unique, non-overlapping directory naming context that is stored on one directory server.

**partner application**
An Oracle Application Server application or non-Oracle application that delegates the authentication function to the OracleAS Single Sign-On server. This type of application spares users from reauthenticate by accepting mod_osso headers.

**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#1**
The Public Key Cryptography Standards (PKCS) are specifications produced by RSA Laboratories. PKCS#1 provides recommendations for the implementation of public-key cryptography based on the RSA algorithm, covering the following aspects: cryptographic primitives; encryption schemes; signature schemes; ASN.1 syntax for representing keys and for identifying the schemes.

**PKCS#5**
The Public Key Cryptography Standards (PKCS) are specifications produced by RSA Laboratories. PKCS#5 provides recommendations for the implementation of password-based cryptography.

**PKCS#7**
The Public Key Cryptography Standards (PKCS) are specifications produced by RSA Laboratories. PKCS #7 describes general syntax for data that may have cryptography applied to it, such as digital signatures and digital envelopes.

**PKCS#8**
The Public Key Cryptography Standards (PKCS) are specifications produced by RSA Laboratories. PKCS #8 describes syntax for private key information, including a private key for some public key algorithms and a set of attributes. The standard also describes syntax for encrypted private keys.

**PKCS#10**
The Public Key Cryptography Standards (PKCS) are specifications produced by RSA Laboratories. PKCS #10 describes syntax for a request for certification of a public key, a name, and possibly a set of attributes.

**PKCS#12**
The Public Key Cryptography Standards (PKCS) are specifications produced by RSA Laboratories. PKCS #12 describes a transfer syntax for personal identity information, including private keys, certificates, miscellaneous secrets, and extensions. Systems (such as browsers or operating systems) that support this standard allow a user to import, export, and exercise a single set of personal identity information—typically in a format called a **wallet**.

**PKI**
See public key infrastructure (PKI).

**plaintext**
Plaintext is readable data prior to a transformation to ciphertext using encryption, or readable data that is the result of a transformation from ciphertext using decryption.

**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.

**policy precedence**
In Oracle Application Server Certificate Authority (OCA), policies are applied to incoming requests in the order that they are displayed on the main policy page. When the OCA policy processor module parses policies, those that appear toward the top of the policy list are applied to requests first. Those that appear toward the bottom of the
list are applied last and take precedence over the others. Only enabled policies are applied to incoming requests.

**policy.properties**
A multipurpose configuration file for Oracle Application Server Single Sign-On that contains basic parameters required by the single sign-on server. Also used to configure advanced features of OracleAS Single Sign-On, such as multilevel authentication.

**POSIX**
Portable Operating System Interface for UNIX. A set of programming interface standards governing how to write application source code so that the applications are portable between operating systems. A series of standards being developed by the Internet Engineering Task Force (IETF).

**POST Profile**
An authentication method whereby login credentials are submitted within the body of the login form.

**predicates**
In Oracle Application Server Certificate Authority (OCA), a policy predicate is a logical expression that can be applied to a policy to limit how it is applied to incoming certificate requests or revocations. For example, the following predicate expression specifies that the policy in which it appears can have a different effect for requests or revocations from clients with DN's that include "ou=sales,o=acme,c=us":

```
Type=="client" AND DN=="ou=sales,o=acme,c=us"
```

**principal**
One of the three primary roles defined in the identity federation protocols supported by OSFS. The other roles are identity provider and service provider.

A principal is any entity capable of using a service and capable of acquiring a federated identity. Typically, a principal is a person or user, or a system entity whose identity can be authenticated.

**primary node**
In an Oracle Application Server Cold Failover Cluster (Identity Management), the cluster node on which the application runs at any given time.

See also: secondary node.

**private key**
A private key is the secret key in a public/private key pair used in public key cryptography. An entity uses its private key to decrypt data that has been encrypted with its public key. The entity can also use its private key to create digital signatures. The security of data encrypted with the entity’s public key as well as signatures created by the private key depends on the private key remaining secret.

**private key cryptography**
See symmetric cryptography.

**profile**
See directory integration profile.
Project Liberty
See Liberty Alliance.

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.

provisioning
The process of providing users with access to applications and other resources that may be available in an enterprise environment.

provisioning agent
An application or process that translates Oracle-specific provisioning events to external or third-party application-specific events.

provisioning integration profile
A special kind of directory integration profile that describes the nature of provisioning-related notifications that Oracle Directory Integration Platform sends to the directory-enabled applications.

proxy server
A server between a client application, such as a Web browser, and a real server. It intercepts all requests to the real server to see if it can fulfill the requests itself. If not, it forwards the request to the real server. In OracleAS Single Sign-On, proxies are used for load balancing and as an extra layer of security.

See also: load balancer.

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
A public key is the non-secret key in a public/private key pair used in public key cryptography. A public key allows entities to encrypt data that can only then be decrypted with the public key's owner using the corresponding private key. A public key can also be used to verify digital signatures created with the corresponding private key.

public key certificate
See certificate.

public key cryptography
Public key cryptography (also known as asymmetric cryptography) uses two keys, one public and the other private. These keys are called a key pair. The private key must be kept secret, while the public key can be transmitted to any party. The private key and the public key are mathematically related. A message that is signed by a private key can be verified by the corresponding public key. Similarly, a message encrypted by the
public key can be decrypted by the private key. This method ensures privacy because only the owner of the private key can decrypt the message.

**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 key infrastructure (PKI)**

A public key infrastructure (PKI) is a system that manages the issuing, distribution, and authentication of public keys and private keys. A PKI typically comprises the following components:

- A **Certificate Authority (CA)** that is responsible for generating, issuing, publishing and revoking digital certificates.
- A **Registration Authority (RA)** that is responsible for verifying the information supplied in requests for certificates made to the CA.
- A directory service where a **certificate** or **certificate revocation list (CRL)** gets published by the CA and where they can be retrieved by relying third parties.
- Relying third parties that use the certificates issued by the CA and the public keys contained therein to verify digital signatures and encrypt data.

**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.

**RC2**

Rivest Cipher Two (RC2) is a 64-bit **block cipher** developed by Ronald Rivest for RSA Security, and was designed as a replacement for **Data Encryption Standard (DES)**.

**RC4**

Rivest Cipher Four (RC4) is a **stream cipher** developed by Ronald Rivest for RSA Security. RC4 allows variable key lengths up to 1024 bits. RC4 is most commonly used to secure data communications by encrypting traffic between Web sites that use the **Secure Sockets Layer (SSL)** protocol.

**RDN**

See **relative distinguished name (RDN)**.

**readable data**

Data prior to a transformation to ciphertext via encryption or data that is the result of a transformation from ciphertext via decryption.

**realm**

See **identity management realm**.

**realm search base**

An attribute in the **root Oracle Context** that identifies the entry in the **directory information tree (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.

Registration Authority (RA)
The Registration Authority (RA) is responsible for verifying and enrolling users before a certificate is issued by a Certificate Authority (CA). The RA may assign each applicant a relative distinguished value or name for the new certificate applied. The RA does not sign or issue certificates.

registry entry
An entry containing runtime information associated with invocations of Oracle Internet 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.

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.

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.

replica
Each copy of a naming context that is contained within a single server.

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.

RFC
The Internet Request For Comments (or RFC) documents are the written definitions of the protocols and policies of the Internet. The Internet Engineering Task Force (IETF) facilitates the discussion, development, and establishment of new standards. A standard is published using the RFC acronym and a reference number. For example, the official standard for e-mail is RFC 822.
root CA
In a hierarchical public key infrastructure (PKI), the root Certificate Authority (CA) is the CA whose public key serves as the most trusted datum for a security domain.

root directory specific entry (DSE)
An entry storing operational information about the directory. The information is stored in a number of attributes.

root DSE
See root directory specific entry (DSE).

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.

RSA
RSA is a public key cryptography algorithm named after its inventors (Rivest, Shamir, and Adelman). The RSA algorithm is the most commonly used encryption and authentication algorithm and is included as part of the Web browsers from Netscape and Microsoft, and many other products.

RSAES-OAEP
The RSA Encryption Scheme - Optimal Asymmetric Encryption Padding (RSAES-OAEP) is a public key encryption scheme combining the RSA algorithm with the OAEP method. Optimal Asymmetric Encryption Padding (OAEP) is a method for encoding messages developed by Mihir Bellare and Phil Rogaway.

S/MIME
See Secure/Multipurpose Internet Mail Extension (S/MIME).

SAML

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 (Identity Management), the cluster node to which an application is moved during a failover.

See also: primary node.
secret key
A secret key is the key used in a symmetric algorithm. Since a secret key is used for both encryption and decryption, it must be shared between parties that are transmitting ciphertext to one another but must be kept secret from all unauthorized entities.

secret key cryptography
See symmetric cryptography.

Secure Hash Algorithm (SHA)
Secure Hash Algorithm (SHA) is a hash function algorithm that produces a 160-bit message digest based upon the input. The algorithm is used in the Digital Signature Standard (DSS). With the introduction of the Advanced Encryption Standard (AES) which offers three key sizes: 128, 192 and 256 bits, there has been a need for a companion hash algorithm with a similar level of security. The newer SHA-256, SHA-284 and SHA-512 hash algorithms comply with these enhanced requirements.

Secure Sockets Layer (SSL)
Secure Sockets Layer (SSL) is a protocol designed by Netscape Communications to enable encrypted, authenticated communications across networks (such as the Internet). SSL uses the public key encryption system from RSA, which also includes the use of a digital certificate. SSL provides three elements of secure communications: confidentiality, authentication, and integrity.

SSL has evolved into Transport Layer Security (TLS). TLS and SSL are not interoperable. However, a message sent with TLS can be handled by a client that handles SSL.

Secure/Multipurpose Internet Mail Extension (S/MIME)
Secure/Multipurpose Internet Mail Extension (S/MIME) is an Internet Engineering Task Force (IETF) standard for securing MIME data through the use of digital signatures and encryption.

Security Assertions Markup Language (SAML)
An XML-based framework which defines mechanisms for exchanging security information about a subject by making assertions about the subject that are used to make access control decisions. SAML enables the exchange of authentication and authorization information between identity providers and service providers who otherwise may not be able to interoperate.

SAML 2.0 is a major revision of the standard which updates SAML 1.1 and combines input from both Shibboleth and Liberty ID-FF specifications. A key aspect of SAML 2.0 is the ability for two sites to establish and maintain an identifier for a user, with that user’s cooperation. Additional features include privacy mechanisms and support for global logout.

security token
In the Liberty protocol, refers to a set of security information that represents and substantiates a claim.

server certificate
A certificate that attests to the identity of an organization that uses a secure Web server to serve data. A server certificate must be associated with a public/private key pair issued by a mutually trusted Certificate Authority (CA). Server certificates are required for secure communications between a browser and a Web server.
service provider
These are organizations recognized by the members of a circle of trust as the entities that provide Web-based services to users. Service providers enter into partnerships with other service providers and identity providers with the goal of providing their common users with secure single sign-on between all parties of the federation.

service time
The time between the initiation of a request and the completion of the response to the request.

session key
A secret key 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.

Signed Public Key And Challenge (SPKAC)
Signed Public Key And Challenge (SPKAC) is a proprietary protocol used by the Netscape Navigator browser to request certificates.

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)
Simple Authentication and Security Layer (SASL) is a method for adding authentication and authorization capabilities to application protocols. SASL provides a security layer between the protocol and the connection, so that users can be authenticated to a server. A security layer can also be negotiated to protect subsequent protocol interactions.

Simple Object Access Protocol (SOAP)
Simple Object Access Protocol (SOAP) is an XML-based protocol that defines a framework for exchanging messages between systems over the Internet. A common protocol for Web Services, SOAP is used with transport protocols such as HTTP and FTP. A SOAP message consists of three parts — an envelope that describes the
message and how to process it, a set of encoding rules for expressing instances of application-defined datatypes, and a convention for representing remote procedure calls and responses.

**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.

**single sign-off**

The process by which you terminate an OracleAS Single Sign-On session and log out of all active partner applications simultaneously. You can do this by logging out of the application that you are working in.

**single sign-on (SSO)**

A process or system that enables a user to access multiple computer platforms or application systems after being authenticated only once.

**single sign-on SDK**

Legacy APIs to enable OracleAS Single Sign-On partner applications for single sign-on. The SDK consists of PL/SQL and Java APIs as well as sample code that demonstrates how these APIs are implemented. This SDK is now deprecated and **mod_osso** is used instead.

**single sign-on server**

Program logic that enables users to log in securely to single sign-on applications such as expense reports, mail, and benefits.

**SLAPD**

Standalone LDAP daemon. An LDAP directory server service that is responsible for most functions of a directory except replication.

**slave**

See **consumer**.

**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.

**SOAP**

See **Simple Object Access Protocol (SOAP)**.

**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.
SPKAC
See Signed Public Key And Challenge (SPKAC).

sponsor node
In replication, the node that is used to provide initial data to a new node.

SSL
See Secure Sockets Layer (SSL).

SSO
See single sign-on (SSO).

stream cipher
Stream ciphers are a type of symmetric algorithm. A stream cipher encrypts in small units, often a bit or a byte at a time, and implements some form of feedback mechanism so that the key is constantly changing. RC4 is an example of a stream cipher.

See also: block cipher.

subACLSubentry
A specific type of subentry that contains access control list (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 CA
In a hierarchical public key infrastructure (PKI), the subordinate Certificate Authority (CA) is a CA whose certificate signature key is certified by another CA, and whose activities are constrained by that other CA.

subordinate reference
A knowledge reference pointing downward in the directory information tree (DIT) to a naming context that starts immediately below an entry.

subschema DN
The list of directory information tree (DIT) areas having independent schema definitions.

subSchemaSubentry
A specific type of subentry containing schema information.
subtree
A section of a directory hierarchy, which is also called a directory information tree (DIT). The subtree typically starts at a particular directory node and includes all subdirectories and objects below that node in the directory hierarchy.

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.

success URL
When using Oracle Application Server Single Sign-On, the URL to the routine responsible for establishing the session and session cookies for an application.

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 directory system agent (DSA) that holds a naming context higher in the directory information tree (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.

symmetric algorithm
A symmetric algorithm is a cryptographic algorithm that uses the same key for encryption and decryption. There are essentially two types of symmetric (or secret key) algorithms — stream ciphers and block ciphers.

symmetric cryptography
Symmetric cryptography (or shared secret cryptography) systems use the same key to encipher and decipher data. The problem with symmetric cryptography is ensuring a secure method by which the sender and recipient can agree on the secret key. If a third party were to intercept the secret key in transit, they could then use it to decipher anything it was used to encipher. Symmetric cryptography is usually faster than
asymmetric cryptography, and is often used when large quantities of data need to be exchanged. **DES**, **RC2**, and **RC4** are examples of symmetric cryptography algorithms.

**symmetric key**
See **secret key**.

**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.

**think time**
The time the user is not engaged in actual use of the processor.

**third-party access management system**
Non-Oracle single sign-on system that can be modified to use OracleAS Single Sign-On to gain access to Oracle Application Server applications.

**throughput**
The number of requests processed by Oracle Internet Directory for each unit of time. This is typically represented as "operations per second."

**Time Stamp Protocol (TSP)**
Time Stamp Protocol (TSP), as specified in RFC 3161, defines the participating entities, the message formats, and the transport protocol involved in time stamping a digital message. In a TSP system, a trusted third-party Time Stamp Authority (TSA) issues time stamps for messages.

**TLS**
See **Transport Layer Security (TLS)**.

**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.

**Triple Data Encryption Standard (3DES)**
Triple Data Encryption Standard (3DES) is based on the **Data Encryption Standard (DES)** algorithm developed by IBM in 1974, and was adopted as a national standard in 1977. 3DES uses three 64-bit long keys (overall key length is 192 bits, although actual key length is 56 bits). Data is encrypted with the first key, decrypted with the second key, and finally encrypted again with the third key. This makes 3DES three times slower than standard DES but also three times more secure.
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, trusted certificates come from a Certificate Authority (CA) you trust to issue user certificates.

trustpoint
See trusted certificate.

TSP
See Time Stamp Protocol (TSP).

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.

URI
Uniform Resource Identifier (URI). A way to identify any point of content on the Web, whether it be a page of text, a video or sound clip, a still or animated image, or a program. The most common form of URI is the Web page address, which is a particular form or subset of URI called a URL.

URL
Uniform Resource Locator (URL). The address of a file accessible on the Internet. The file can be a text file, HTML page, image file, a program, or any other file supported by HTTP. The URL contains the name of the protocol required to access the resource, a domain name that identifies a specific computer on the Internet, and a hierarchical description of the file location on the computer.

URLC token
The OracleAS Single Sign-On code that passes authenticated user information to the partner application. The partner application uses this information to construct the session cookie.

user name mapping module
A OracleAS Single Sign-On Java module that maps a user certificate to the user's nickname. The nickname is then passed to an authentication module, which uses this nickname to retrieve the user's certificate from the directory.

user search base
In the Oracle Internet Directory default directory information tree (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).

**UTF-16**
16-bit encoding of Unicode. The Latin-1 characters are the first 256 code points in this standard.

**verification**
Verification is the process of ensuring that a given digital signature is valid, given the public key that corresponds to the private key purported to create the signature and the data block to which the signature purportedly applies.

**virtual host**
A single physical Web server machine that is hosting one or more Web sites or domains, or a server that is acting as a proxy to other machines (accepts incoming requests and reroutes them to the appropriate server).

In the case of OracleAS Single Sign-On, virtual hosts are used for load balancing between two or more OracleAS Single Sign-On servers. They also provide an extra layer of security.

**virtual host name**
In an Oracle Application Server Cold Failover Cluster (Identity Management), the host name corresponding to a particular virtual IP address.

**virtual IP address**
In an Oracle Application Server Cold Failover Cluster (Identity Management), 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.

**wait time**
The time between the submission of the request and initiation of the response.

**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.

**Wallet Manager**
See Oracle Wallet Manager.

**Web service**
A Web service is application or business logic that is accessible using standard Internet protocols, such as HTTP, XML, and SOAP. Web Services combine the best aspects of component-based development and the World Wide Web. Like components, Web
Services represent black-box functionality that can be used and reused without regard to how the service is implemented.

**Web Services Description Language (WSDL)**

Web Services Description Language (WSDL) is the standard format for describing a Web service using **XML**. A WSDL definition describes how to access a Web service and what operations it will perform.

**WSDL**

See **Web Services Description Language (WSDL)**.

**WS-Federation**

Web Services Federation Language (WS-Federation) is a specification developed by Microsoft, IBM, BEA, VeriSign, and RSA Security. It defines mechanisms to allow **federation** between entities using different or like mechanisms by allowing and brokering trust of identities, attributes, and authentication between participating **Web services**.

See also: **Liberty Alliance**.

**X.500**

X.500 is a standard from the International Telecommunication Union (ITU) that defines how global directories should be structured. X.500 directories are hierarchical with different levels for each category of information, such as country, state, and city.

**X.509**

X.509 is the most widely used standard for defining digital certificates. A standard from the International Telecommunication Union (ITU), for hierarchical directories with authentication services, used in many **public key infrastructure (PKI)** implementations.

**XML**

Extensible Markup Language (XML) is a specification developed by the World Wide Web Consortium (W3C). XML is a pared-down version of Standard Generalized Mark-Up Language (SGML), designed especially for Web documents. XML is a metalanguage (a way to define tag sets) that allows developers to define their own customized markup language for many classes of documents.

**XML canonicalization (C14N)**

This is a process by which two logically equivalent XML documents can be resolved to the same physical representation. This has significance for digital signatures because a signature can only verify against the same physical representation of the data against which it was originally computed. For more information, see the W3C’s XML Canonicalization specification.
abstract object classes, 3-13
superclasses of, 11-4
top, 3-12
access
exclusionary, 18-13
granting
  by using command-line tools, 18-35
  by using Oracle Directory Manager, 18-14
entry-level, by using command-line
tools, 18-37
entry-level, by using Oracle Directory
Manager, 18-24
kinds, 18-8
level requirements for LDAP operations, 18-10
object, 18-6
operations, 18-8
rights, setting by using Oracle Directory
Manager, 18-17, 18-23
selecting, by DN, 18-37
subject, 18-7
unspecified, 18-10, 18-23
violation event, 14-10
access control
and authorization, 3-15
conceptual discussion, 16-3
default, 21-3
defined, 3-15
directive format. See ACI directive format
management constructs, 18-2
managing, 18-1
  by using command-line tools, 18-35
  by using Oracle Directory Manager, 18-14
overview, 2-7
policies
  conflicting, 18-2
  inheriting, 18-2
policy administration, overview, 18-1
prescriptive, 18-2
setting, by using wildcards, 18-37
access control information (ACI)
  attributes, 16-3
  components, 18-6
  directives, format
  items
format, C-1
syntax, C-1
more than one for the same subject, 18-12
object of directives, 18-6
subject of directives, 18-7
access control lists (ACLs), 3-7, 16-3
directives, within entries, 18-2
evaluation
  for groups, 18-13
  precedence rules, 18-11
for groups, 18-13
how it works, 18-10
modification, 14-10
precedence
  rules, 18-11
within subtrees, 18-2
Access Control Management pane, in Oracle
Directory Manager, A-4
access control policy points (ACPs), 18-2, 18-16
adding
  by using ldapmodify, 18-36
  by using Oracle Directory Manager, 5-4, 18-16
  by using the ACP Creation Wizard of Oracle
Directory Manager, 18-18
administering, by using Oracle Directory
Manager, 5-7
configuring display of, in Oracle Directory
Manager, 18-14
creating by using ACP Creation Wizard, 18-18
Creation Wizard, 18-18
defined
groups, 18-3
multiple, 18-2
viewing, 18-15
  by using Oracle Directory Manager, 18-15
viewing, by using Oracle Directory
Manager, 18-15
accounts
enabling and disabling
  by using command-line tools, 19-11
  by using Oracle Internet Directory Self-Service
Console, 19-12
unlocking
  by using command-line tools, 19-11
  by using Oracle Internet Directory Self-Service
Console, 19-12
realm administrator, 23-11
super user, 19-4
ACI. See access control information (ACI)
ACL. See access control lists (ACLs)
ACP groups, 18-3
ACP. See access control policy points (ACPs)
ACPs. See access control policy points (ACPs)
active server instances
modifying configuration set entries in, 7-3
viewing, 7-3, 7-11
added_object_constraint filter, 18-36
added-object-constraint, in access control, 18-8
administration tools, 8-7
Catalog Management Tool (catalog), 5-10
command-line, 2-6, 5-8
Human Intervention Queue Manipulation Tool, 5-12
ldapadd, 8-7
ldapdelete, 8-7
ldapmodify, 8-8
OID Database Password Utility (oidpasswd), 5-13
OID Database Statistics Tool (oidstats.sql), 5-12
OID Migration Tool, 5-12
OID Reconciliation Tool, 5-12
Oracle Directory Manager, 5-1
Replication Environment Management Tool, 5-12
Advanced Replication
configuring
by using Replication Management Tool, 30-9
agents
log file location, 14-1
alias entries
adding, 7-13
dereferencing, 7-12, 7-13
messages, 7-16
modifying, 7-15
searching directory with, 7-14
ANALYZE function of DBMS_STATS package, 25-2
anonymous authentication, 16-4, A-2
anonymous login, A-2
Application Server Control
starting directory server instance, 14-21
stopping directory server instance, 14-21
viewing user logon session information, 14-22
application-specific repositories
migrating data from, 27-8
Apply button, in Oracle Directory Manager, 5-3
architecture
Oracle Internet Directory, 2-5, 3-1
Oracle Internet Directory Server Manageability framework, 14-14
ASR Agreement tab page, in Oracle Directory Manager, A-13
attribute alias, 11-20
attribute options, 3-11
adding
by using ldapmodify, 8-9
by using Oracle Directory Manager, 8-6
conceptual discussion, 3-11
deleting by using Oracle Directory Manager, 8-7, 8-9
language codes, 3-11
managing
by using command line tools, 8-9
by using Oracle Directory Manager, 8-6
modifying by using Oracle Directory Manager, 8-6
searching for by using ldapsearch, 8-9
attribute uniqueness
about, 10-1
constraint entries, 10-1
descriptions
location of, 10-5
known limitations, 10-9
managing, 10-5
managing by using command-line tools, 10-6
managing by using Oracle Directory Manager, 10-5
rules for creating, 10-2
attributes
adding, 11-9
by using ldapmodify, 11-14
by using Oracle Directory Manager, 11-11
guidelines for, 11-9
as DN, 8-3
as metadata in schema, 11-1
attribute options, 8-9
adding by using ldapmodify, 8-9
adding by using Oracle Directory Manager, 8-6
costnceptual discussion, 3-11
deleting by using Oracle Directory Manager, 8-7, 8-9
managing by using command line tools, 8-9
managing by using Oracle Directory Manager, 8-6
modifying by using Oracle Directory Manager, 8-6
base schema
deleting, 11-10
modifying, 11-9
costnceptual discussion, 3-11
deleting by using Oracle Directory Manager, 5-4
guidelines for, 11-10
determined by object classes, 11-2
ditcontentrule, 11-17
dropping indexes, 11-13
extending number of
by using auxiliary object classes, 11-17
by using content rules, 11-17
for existing entries, 11-17
prior to creating entries, 11-16
for a specific entry
viewing by using Oracle Directory Manager, 8-3
for which data exists
indexing, 11-16
for which no directory data exists
indexing, 11-15
in base schema, 11-9
in top, 3-13
indexed
viewing, 11-13
indexes, created by bulkload, 9-5
indexing, 11-16
by using Catalog Management tool
by using command-line tools, 11-15
by using Oracle Directory Manager, 11-13
when you create them, 11-13
information, kinds of, 3-9
inheritance of, 11-2
jpegPhotos, 3-10, 8-8
kinds of information in, 3-9
labeledURI, 13-3, 13-9, 13-11
making available for searches, 11-13
managing, 11-9
by using command-line tools, 11-13
by using Oracle Directory Manager, 11-8, 11-10
overview, 11-9
managing by using command-line tools, 11-13
mandatory, 3-12, 8-5, 11-2
in a user entry, 27-10
matching rules, 3-11
modifying
by using ldapmodify, 8-8
by using ldapmodifymt, 8-8
by using Oracle Directory Manager, 8-6, 11-12
guidelines for, 11-9
rules for, 11-9
using ldapmodify, 11-14
multivalued, 3-10, 18-3
converting to single-valued, 11-9
null values in, 11-2
objectclass, 14-9
objects associated with an ACI, 18-6
operational, 7-7
optional, 3-12, 11-2
options, 3-11
language codes., 3-11
orclauditlevel, 14-11
orclauditmessage, 14-9
orclauditoc, 14-9
orcleventtime, 14-9
orcleventtype, 14-9
orclopresult, 14-9
orclsequence, 14-9, 14-10
orclskewedattribute, 25-11
orcluserdn, 14-9
organization, 3-10
organizationalUnitName, 3-10
redefining mandatory, 11-4
ref, 8-10
removing from object classes, 11-4
rules
for adding, 11-9
for deleting, 11-10
for modifying, 11-9
searching for, by using Oracle Directory
Manager, 11-10
single-valued, 3-10
converting to multivalued, 11-9
skewed, optimizing searches for, 25-10
sn, 3-10
specifying as mandatory or optional, 11-3
surname, 3-10
text, 3-11
modifying, 11-9
text type
selecting, 11-26
textual
cannot modify, 11-9
selecting, 11-26
system operational, 7-7
types, 3-9
usercertificate, G-1
values, 3-9
viewing, 8-3
Attributes tab page, in Oracle Directory
Manager, A-19
audit level, 14-10
modifying, 14-12
setting, 14-11
by using ldapmodify, 14-12
by using Oracle Directory Manager, 14-11
audit log, 14-8
container object, 14-13
default configuration, 14-8
entries
in the DIT, position of, 14-10
position in DIT
searching, 14-9
searching for, 14-12
searching for by using ldapsearch, 14-13
searching for by using Oracle Directory
Manager, 14-12
structure, 14-9
viewing, 14-8
events
access violation, 14-10
ACL modification, 14-10
add, 14-10
adding, 14-10
bind, 14-10
deleting, 14-11
DSE modification, 14-10
modify, 14-11
modifyDN, 14-11
modifying, 14-11
replication login, 14-10
schema element, add/replace, 14-10
schema element, delete, 14-10
selected, 14-11
super user login, 14-10
user password modification, 14-10
garbage collector, 26-2
purging, 14-13
queries, 14-8
sample, 14-10
structure of entries, 14-9
using, 14-8
auditable events, 14-10
auditing selected events, 14-11
authenticated access, by using SSL, 2-7
authentication, 16-4
anonymous, 16-4, A-2
certificate, 16-4
conceptual discussion, 16-4
defined, 3-15
direct
options, 16-4
external, 16-7, 34-1
SASL, 16-4
in a typical directory operation, 3-7
indirect, 16-5
through a RADIUS server, 16-5
native, 34-1
Oracle directory replication server, 29-18
password-based, 16-4, A-2
PKI, 16-2
SASL, 16-4
SASL mechanism
external authentication, 16-4
MD5 Digest, 16-4
simple, 2-7, 16-4, A-2
Simple Authentication and Security Layer (SASL), 16-4
SSL
defined, 16-4
for Oracle Directory Manager, A-3
no, A-3
server only, A-3
three levels, 2-7
through a middle tier, 16-5
Authentication Choice list, in Oracle Directory Manager, A-4
Authentication Services Group, 21-12
authorization, 3-15, 16-3
automated resolution of conflicts, 29-26
auxiliary object classes, 3-13, 11-4
extending number of attributes by using, 11-17
average latency, 25-1

B
backup and recovery strategies, failover, 22-4
backup and restore, 15-1
base schema
attributes, 11-9
deleting, 11-10
modifying, 11-9
object classes
modifying, 11-4
base search, 8-2
batching line-mode commands, 11-7
Begins With filter, in Oracle Directory Manager, A-18
bind event, 14-10
bind mode, 18-8
binding, 3-7
BSTAT/ESTAT scripts, 25-5
buffer caches, size, 25-5
bulk loading failure, 9-5
bulkdelete, 5-11
and Globalization Support, D-9
log file location, 14-1
bulkload, 5-11, 9-4, 9-5
and Globalization Support, D-8
check mode, performing on LDIF files, 27-8
creating indexes, 9-5
load option, 9-5
log file location, 14-1
bulkmodify, 5-11
and Globalization Support, D-9
log file location, 14-1
By Whom tab page, in Oracle Directory Manager, A-4

C
C API, 3-7
cache, entry, 25-8
cache, metadata, 3-4
caching
client-side referral, 8-12
Cancel button, in Oracle Directory Manager, 5-3
capacity planning, 22-5, 22-6, 24-1
I/O subsystem, 24-5
network requirements, 24-10
overview, 24-1
catalog entry, 3-5
Catalog Management Tool (catalog), 11-16
log file location, 14-1
Catalog Management Tool (catalog.sh), 5-10
catalog. See Catalog Management tool.
cataloged attributes
oracleventtype, 14-9
orcluserdn, 14-9
certificate authentication, 16-4
change log
purging, in multimaster replication, 26-5
change log garbage collection
troubleshooting, L-20
Change Log window, in Oracle Directory Manager, A-17
change logs, 3-18, 29-5
and directory replication, 29-19
change number-based purging, 26-5
garbage collector, 26-2
in replication, 2-7, 29-19, 29-23
purging, 26-5
methods, 26-5
time-based purging, 26-5
change number-based purging, 26-5
change retry count, setting, A-12
cipher suites
SSL, 17-1
SSL, supported, 17-1
SSL_RSA_WITH_3DES_EDE_CBC_SHA, 17-1
SSL_RSA_WITH_NULL_MD5, 17-1
SSL_RSA_WITH_NULL_SHA, 17-1
SSL_RSA_WITH_RC4_128_SHA, 17-1
client-side referral caching, how it works, 8-12
command line tools
adding configuration set entries, 3-6, 8-7
Catalog Management Tool, 11-13
comparing attribute values, 8-7
for managing entries, 8-7
indexing, 11-13, 11-16
ldapadd, 8-7
ldapaddmt, 8-7
ldapdelete, 8-7
ldapmodi, 8-8
managing
attributes, 11-13
entries, 8-7
modifying configuration set entries, 8-8
overview, 8-7
setting Globalization Support, D-5
common entry, defined, 3-5
Common Group Attributes Group, 21-14
Common User Attributes Group, 21-14
commonName attribute, 3-10
comparing
attribute values, 8-7
two objects, 5-4
component deployment and administration
delegation, 21-8
components
of a directory server, 3-2
of Oracle Internet Directory, 2-6
compound RDNs
oidcmprec limitations, 31-18
concurrent database connections, 25-8
configSets, 3-6
configuration parameters
modifying, 3-6
Oracle directory replication server
location, 31-1
configuration set entries, 3-6
adding, 3-6, 7-1, 7-5
by using command line tools, 8-7
by using command-line tools, 3-6
by using Oracle Directory Manager, 7-3
changing, 7-6
deleting, 7-1
by using ldapmodi, 7-6
by using Oracle Directory Manager, 7-3, 7-5
for replication server, 31-1
LDIF files, 7-5
managing, 1-4, 7-1
by using command line tools, 7-5
by using Oracle Directory Manager, 7-3
preliminary considerations, 7-2
modifying, 3-6, 7-1, 7-2
by using ldapmodi, 7-6
by using Oracle Directory Manager, 7-3, 7-5
in an active server instance, 7-3
multiple, 17-3
overriding user-specified, 1-14
SSL parameters in, 17-1
using different, 7-2
viewing, 7-3
configuration set location, A-25
Configuration Sets General tab page, in Oracle Directory Manager, A-24
conflict resolution, in replication, 29-25
conflicting access control policies, 18-2
precedence, rules for resolving, 18-2
conflicts, replication
automated resolution of, 29-26
manual resolution of, 30-37
resolution, 18-11, 29-25
resolving manually, 30-37
typical causes of, 29-26
CONNECT BY assertions, in dynamic groups, 13-4
Connect/Disconnect button in Oracle Directory Manager, 5-5
connecting
to a directory server, 5-2
in a typical directory operation, 3-7
to additional directory servers, 5-5
to multiple directory servers, 5-5
connection
pooling, 2-7
connections, LDAP, specifying maximum idle time for, 7-11
constraints, object classes, 3-13
consumers
defined, 3-17, 29-3
containment
of groups, planning, 23-3
of users, planning, 23-3
content access items, 18-27
of an existing ACP, 18-23
Content Rule dialog box, in Oracle Directory Manager, A-23
content rules
defined, 11-17
defined as values of ditcontentrule attribute, 11-17
extending number of attributes by using, 11-17
managing
by using command-line tools, 11-19
by using Oracle Directory Manager, 11-19
rules for creating and modifying, 11-17
schema enforcement when using, 11-18
control, access, 2-7, 18-1
controls
definition, 2-4
converting
auxiliary object classes, 11-4
structural object classes, 11-4
CPUs
configuration, 24-11
in capacity planning, 24-1
power required for various deployment scenarios, 22-7
processing power, 24-11
requirements, 24-11
detailed calculations, 24-12
in capacity planning, 24-11
tuning, 25-2
tuning for Oracle foreground processes, 25-4
usage, 22-8
usage tuning, 25-2
when to tune, 25-3
Create button, in Oracle Directory Manager, 5-5
Create Entry menu item, in Oracle Directory Manager, 5-4
Create Like
adding entries using templates, 8-4
button, in Oracle Directory Manager, 5-5, 8-4
operation, by using Oracle Directory Manager, 5-4
createsTimestamp attribute, 3-10, 27-7
optional in top, 3-13
creatorsName attribute, 3-10, 27-7
optional attribute in top, 3-13
critical events
in Oracle Internet Directory Server Manageability framework, 14-19
levels, 14-19
D
data integrity, 3-15, 16-1
data migration process, 27-2
data privacy, 3-15, 16-1
by using SSL, 2-7
data, updating by using Oracle Directory Manager, 5-5
database
block buffers parameter, 25-7
block size parameter, 25-7
cache size, 22-7
connections, 3-4
concurrent, 25-8
pooling, 2-7
dedicated for directory, 3-3
password, changing, 7-11
queries, optimization of, 25-10
server, 2-5
server error, L-2
tuning, 25-7
DB_BLOCK_BUFFERS, 25-7
DBMS_STATS package, 25-2
debug
log files, viewing, L-14
debug dimension, 14-6
debug logging
levels, 14-5
about, 14-2
setting, 14-5
setting by using OID Control Utility, 14-5
setting by using Oracle Directory Manager, 14-5
levels, setting
by using OID Control Utility, 14-5
by using Oracle Directory Manager, 14-5
log files, viewing, 14-5
debbuging the external authentication plug-in, 34-3
debbuging, limiting to specific operations, 14-6
default identity management realm, 3-25, 23-10
default configuration
access controls, 21-3
default knowledge references (referrals)
configuring, 8-11
default knowledge references (referrals), configuring, 8-11
default port, 5-2
Delegated Administration Services
defined, 3-21
overview, 3-26
delegation
component deployment and administration, 21-8
how it works, 21-1
in an Oracle Application Server environment, 21-2
of privileges for user and group management, 21-3
Delete button, in Oracle Directory Manager, 5-5
deployment
considerations, 22-1
CPU power, 22-7
failover, 22-4
replication, 22-3
tuning, 22-8
partitioning, 22-3
deployment considerations
metadirectory, 22-5
dererferencing alias entries, 7-13
DES40 encryption, 16-2
descriptions of object classes, A-18, A-19
Digest
MD5, 16-4
directories
access control, 2-7, 18-1
application, migrating data from, 27-8
as read-focused, 2-2
backup and restore, 15-1
contrasted to relational databases, 2-2
database listener, 30-10
declared, 2-1
distributed, 3-16
existing, default directory structure, 27-1
expanding role of, 2-1, 22-1
location-independent, 2-2
multimaster replication groups (DRGs) installing, 30-6
online expanding role of, 2-1
partitioned, 3-18
password, changing, 7-9
planning structure of, 23-3
read-focused, 2-2
replication groups (DRGs), 29-20, 30-7
and replication agreements, 29-20
configuring, 30-6
schema
managing, 11-1
overview, 11-1
small
backing up and restoring, 15-1
special purpose, 2-2
directory information tree (DIT), 3-7
audit log entries in
browsing, 8-2
default, 23-10
planning for identity management, 23-1
directory integration platform server
log file location, 14-1
directory metadata
defined, 3-4
Directory Replication Group, 30-5
directory replication server, 2-6, 3-3
authentication, 29-18
configuration set entries, 31-1
log file location, 14-1
directory schema, 11-1
defined, 3-4
managing, 11-1
directory servers, 2-6, 3-3
adding, A-2
changing parameters in an active instance, 7-3
configuration set entries, 7-1
connecting to, 5-2, 5-5, A-2
by using Oracle Directory Manager, 5-5
in a typical directory operation, 3-7
connecting to additional, 5-5
connecting to one on a different host, A-2
connecting to, by using Oracle Directory Manager, 5-4
disconnecting from, using Oracle Directory Manager, 5-6
disconnecting, by using Oracle Directory Manager, 5-4, 5-6
discovery by using the Domain Name System (DNS), 7-17
locating in a distributed environment, 7-16
log file location, 14-2
modifying, A-2
modifying configuration set entries, 7-6
multimaster replication between, 2-7
parameters
configuring by using command-line tools, 1-4
processes, 3-4
multiple, 3-4
restarting, 7-3
restarting, by using the Application Server Control, 14-21
shared server, 2-7
specifying host, A-2
starting
by using Application Server Control, 14-21
with different configuration, L-14
static discovery by using ldap.ora, 7-16
stopping
by using Application Server Control, 14-21
terminating, 1-4
user logon session information
viewing by using Application Server Control, 14-22
using different configuration set entries, 7-2
viewing information, 7-11
directory usage patterns, learning, 24-2
DirectoryReplicationGroupDSAs, 31-6
Disconnect
button, in Oracle Directory Manager, 5-4
menu item, in Oracle Directory Manager, 5-4
disconnecting from directory servers, 5-6
disk space requirements, 24-5
detailed calculations of, 24-6
estimating, 24-5
disk tuning, 25-6
disk usage, 22-8
displaying
a directory entry, 8-1
a subtree, 8-1
distinguished names, 3-7
as attributes, 8-3
components of format
modifying
by using command line tools, 8-8
distributed directories, 3-16, 3-18
locating directory servers in, 7-16
partitioned, 3-16
partitions and replicas, 22-2
partitions, replicas, and high availability, 22-2
replicated, 3-16
DIT. See directory information tree (DIT)
ditcontentrule attribute, 11-17
dns. See distinguished names.
Domain Name System (DNS)
registering a directory server with, 7-18
server discovery by using, 7-17
DRG, 30-5
Drop Index
button, 5-5
menu item, 5-4
DSE modification event, 14-10
duration of a search, specifying, 8-2, 14-13
dynamic directory server discovery, 7-16, 7-17
dynamic groups, 13-2
entries
managing by using command-line tools, 13-11
managing by using Oracle Directory Manager, 13-9
schema elements for creating, 13-3
dynamic password verifiers
troubleshooting, L-21
-E argument in Globalization Support, D-6
Edit
button, in Oracle Directory Manager, 5-5
menu item, in Oracle Directory Manager, 5-4
Edit Plug-in dialog box, in Oracle Directory
Manager, A-10, A-11, A-12
encryption
DES40, 16-2
levels available in Oracle Internet Directory, 16-2
password, 16-7
passwords
UNIX crypt, 20-3, 20-4
RC4_40, 16-2
Encryption Choice list, in Oracle Directory
Manager, A-4
Ends With filter, in Oracle Directory Manager, A-4
entity component, in access control, 18-7
entries
adding
by copying an existing entry, 8-4
by using ldapadd, 8-7
by using ldapaddmt, 8-7
by using Oracle Directory Manager, 8-3
concurrently, 8-7
mandatory attributes, 8-4
optional attributes, 8-4
requires write access to parents, 8-3
alias, dereferencing, 7-12
attributes, viewing, 8-3
audit log, 14-8
searching, 14-9
command-line tools for managing, 8-7
comparing, by using ldapcompare, 8-7
conceptual discussion, 3-7
configuration set, 3-6
creating by using Oracle Directory Manager, 5-4
deleting
by using ldapdelete, 8-7
displaying, 8-1
distinguished names of, 3-7
garbage collector, 26-5
group, 3-10
inheriting attributes, 11-2
loading, 11-3
locating by using distinguished names
managing, 8-1
by using bulk tools, 9-1
by using command line tools, 8-7
by using Oracle Directory Manager, 5-7, 8-1
managing by using command-line tools, 8-7
modifying
by using Oracle Directory Manager, 8-5
naming, 3-7
objects associated with an ACI, 18-6
parent, 11-3
replication naming context container, 29-13
restricting the kinds users can add, 18-17, 18-19,
18-22, 18-25, 18-36
root of search, 8-1
searching
base level, 8-2
by using Oracle Directory Manager, 8-1
one-level, 8-2
specifying search depth, 8-2
subtree level, 8-2
selecting by DN, 18-37
selecting superclass, 8-3
specific, granting access to, A-4
static group
modifying, by using ldapmodify, 13-8, 13-12
statistics collector, 26-5
superclasses, selecting, 8-3
user
adding, by using ldapadd, 8-8
adding, by using Oracle Directory
Manager, 8-4
modifying, 8-8
modifying, by using ldapmodify, 8-8
modifying, by using Oracle Directory
Manager, 8-5
with attribute options
adding by using ldapmodify, 8-9
adding by using Oracle Directory
Manager, 8-6
deleting by using Oracle Directory
Manager, 8-7, 8-9
managing by using command line tools, 8-9
managing by using Oracle Directory
Manager, 8-6
modifying by using Oracle Directory
Manager, 8-6
searching for by using ldapsearch, 8-9
entry
caching, 25-8
enabling, A-26
catalog, defined, 3-5
common, defined, 3-5
password policy, defined, 3-6
password verifier, defined, 3-6
plug-in, defined, 3-6
entry-level access, granting by using Oracle Directory
Manager, 18-24
environment variables, NLS_LANG, D-3
error messages, L-5
30SendPort, L-2
additional, L-5
administration, L-2
database server, L-2
directory server, due to interrupted client
connection, L-2
directory server, due to schema
modifications, L-2
installation, L-1
ORA-1562, L-2
password policies, L-7
returned from Oracle directory server, L-3
sgslnuRead, L-2
standard, L-3
events, auditable, 14-10
Exact Match filter, in Oracle Directory Manager, A-18, A-31
exclusionary access to objects, granting, 18-13
existing ACPs and their ACI directives, modifying, 18-21
Exit menu item, in Oracle Directory Manager, 5-4
explicit hierarchies, 13-5
extensibility, in LDAP Version 3, 2-4
extensibleObject object class, 8-10
external authentication, 16-7
contrasted with native authentication, 34-1
defined, 34-1
plug-in, 34-1
debugging, 34-3
installing, 34-2, 34-4
installing, configuring, and enabling, 34-2
SASL authentication mechanism, 16-4
external repository, storing security credentials in, 34-1

F
failover, 2-7
considerations in deployment, 22-4
failure to apply changes, 29-25
failure tolerance, and replication, 22-4
fan-out replication, 3-17, 29-4
groups, 3-17, 29-4, 29-6
in conjunction with multimaster replication groups, 29-7
LDAP-based, 3-17
features, new, i-xxxix
in Oracle Internet Directory, Release 3.0.1, i-li
release 10g (10.1.2), i-xlxi
release 10g (10.1.3), i-xxxix
release 10g (9.0.4), i-xlxi
release 2.1.1, i-lii
release 3.0.1, i-li
release 9.0.2, i-xlvi
File menu, in Oracle Directory Manager, 5-4
filters
    Begins With, A-18
    Ends With, A-18
    Exact Match, A-18, A-31
    Greater or Equal, A-18, A-31
    in attribute searches, 11-10
    in searches, 3-7, 11-5
    in Oracle Directory Manager, 11-5
    Less or Equal, A-18, A-31
    not null, A-18
    Present, Oracle Directory Manager, A-31
Find Attributes button, in Oracle Directory Manager, 11-10
Find Objects button, in Oracle Directory Manager, 5-5, 11-5
formats, of distinguished names
full replication, 3-17, 29-2
function calls, tracing, 14-6

G
garbage collection framework
    about, 26-1
    components of, 26-1
    how it works, 26-4
    in replication, 26-5
    plug-in, 26-1
Garbage Collector window, in Oracle Directory Manager, A-6
garbage collectors
    audit log, 26-2
    change log, 26-2
    definition, 26-2
    entries for, 26-5
general statistics, 26-2
    health statistics, 26-2
    managing, 26-6
    modifying
        by using command-line tools, 26-7
        by using Oracle Directory Manager, 26-6
    predefined, 26-2
    security and refresh events, 26-2
    system resource events, 26-3
    tombstone, 26-3
general statistics garbage collector, 26-2
Globalization Support, 3-15
bulkdelete, D-9
bulkload, D-8
bulkmodify, D-9
command-line tools, D-5
Java clients, 3-16
ldapadd, D-6
ldapaddmt, D-6
ldapbind, D-6
ldapcompare, D-7
ldapdelete, D-7
ldapmoddn, D-7
ldapmodify, D-6
ldap MODIFYMT, D-7
ldapssearch, D-6
ldifwrite, D-8
managing, D-1
settings for Oracle Internet Directory, D-3
using with Bulk Tools, D-8
with bulkdelete, D-9
with bulkload, D-8
with bulkmodify, D-9
with command-line tools, D-5
with LDIF Files, D-4
with ldifwrite, D-8
Greater or Equal filter, in Oracle Directory Manager, A-18, A-31
group entries, 3-10
adding, 8-5, 13-6
creating
    by using Oracle Directory Manager, 13-6, 13-9
groupOfNames object class, 13-6, 13-7, 13-9, 13-10
groupOfUniqueNames object class, 13-6, 13-9
groups
ACL evaluation for, 18-13
ACP, 18-3
dynamic, 13-2
  managing by using command-line tools, 13-11
  managing by using Oracle Directory Manager, 13-9
schema elements for creating, 13-3
dynamic and static, administration of, 13-1
granting access rights to, 18-4
hierarchical, 13-4
membership
  how directory server computes, 18-4
names and containment, planning, 23-3
privilege, 18-3
defined, 3-5
static, 13-1
  managing by using command-line tools, 13-7
  managing by using Oracle Directory Manager, 13-6
schema elements for creating, 13-1
when to use static or dynamic, 13-5
guest users
definition, 7-9
managing, 7-9
  by using ldapmodify, 7-10
  by using Oracle Directory Manager, 7-9
user name and password, 7-9
guidelines
  for adding attributes, 11-9
  for adding object classes, 11-3
  for deleting attributes, 11-10
  for deleting object classes, 11-5
  for modifying attributes, 11-9
  for modifying object classes, 11-4

H
hashing
  passwords to the directory, 20-1
  protection
    MD4, 20-2
hashing algorithm
  default, 20-2
  userPassword, 20-2
health statistics garbage collector, 26-2
Help
  button, in Oracle Directory Manager, 5-5
  menu item, in Oracle Directory Manager, 5-4
hierarchical groups, 13-4
hierarchies
  explicit, 13-5
  implicit, 13-4
high availability, 2-7, 22-2, 22-4
  considerations, 22-4
Human Intervention Queue Manipulation Tool, 5-12, 30-38

I
identity management, 23-10
  defined, 3-23
Oracle Identity Management infrastructure, 23-1
  planning DIT for, 23-1
  policies, 3-25
realms
  configuring, 23-12
  customizing, 23-12
  default, 3-25
  defined, 3-25
  entry in default DIT, 23-9
  implementation in Oracle Internet Directory, 23-9
  in enterprise deployments, 23-7
  in hosted deployments, 23-8
  multiple in enterprise deployments, 23-8
  planning, 23-5
  single in enterprise deployments, 23-7
realm-specific Oracle Context, 23-10
identity management realms, 23-7
  creating additional, 23-18
  multiple, 23-8
  single, 23-7
idle time, specifying maximum for LDAP connections, 7-11
IETF
  LDAP approval
  implicit hierarchies, 13-4
indexed attributes
  locations, A-27
  oracleventtype, 14-9
  orcluserdn, 14-9
  viewing, 11-13
indexes
  created by bulkload, 9-5
  dropping from attributes, 11-13, 14-9
    by using Oracle Directory Manager, 11-13
inheritance, 3-12
  and access control policies, 18-2
  from superclasses, 11-2
input file, creating, 7-6
installation errors, L-1
installation types
  in multimaster replication group installation, 30-2
  insufficient memory, 25-5
  intelligent client failover, 22-4
  intelligent network level failover, 22-4
  intermediate template file
    in migration from application-specific repositories, 27-8
  internationalization, and LDAP, D-1
Internet Engineering Task Force (IETF). See IETF.
  introduction to LDAP and Oracle Internet Directory, 2-1
I/O subsystem, 24-5
  in capacity planning, 24-1, 24-5
  requirements, 24-4
  sizing, 24-5
  throughput, maximizing, 24-5
iostat utility, 25-2
Java clients, Globalization Support and, 3-16
Java Native Interface, 3-7
jpegPhoto attribute, 3-10, 8-8

knowledge references, 3-19, 3-20, 22-2, 22-3
configuring, 8-10
default
configuring, 8-11
developed, 3-19
managing, 8-10
overview, 3-19
restricting permissions for managing, 3-20
smart
configuring, 8-10
superior, 3-19

labeledURI attribute, 13-3, 13-9, 13-11
language codes, as attribute options, 3-11
latency, average, 25-1
LDAP
add or modify performance, L-9
and internationalization, 3-15
and simplified directory management, 2-3
attributes, common, 3-10
extensibility, 2-4
IETF approval, 2-3
search performance, L-8
security, 2-4
server instances, 3-3
servers, 3-3
managing, 7-1
multithreaded, 2-7
Version 3, 2-3
LDAP connections, specifying maximum idle time for, 7-11
LDAP Data Interchange Format (LDIF), 5-8
LDAP dispatcher
log file location, 14-2
LDAP replica states, H-1
LDAP replication
process, 29-23
ldapadd, 8-7
and Globalization Support, D-6
ldapaddmt, 8-7
and Globalization Support, D-6
LDAP-based partial replication
determining what is to be replicated, 30-33
LDAP-based replica
configuring, 30-22
deleting, 30-33
LDAP-based replication, 3-17, 29-4
configuring, 30-20
replica states, H-1
ldapbind
and Globalization Support, D-6
ldapbind operation, 16-4
ldapcompare, 8-7
and Globalization Support, D-7
LDAP-compliant directories, migrating data from, 27-2
ldapdelete, 8-7
and Globalization Support, D-7
ldapmoddn, 8-8
and Globalization Support, D-7
ldapmodify, 8-8
adding ACPs, 18-36
adding attributes, 11-14
adding entry-level ACIs
adding object classes, 11-7
and Globalization Support, D-6
changing audit level, 14-12
modifying attributes, 11-14
modifying object classes, 11-7
ldapmodifyfmt, 8-8
and Globalization Support, D-7
ldap.ora, 7-16
server discovery by using, 7-16
ldapsearch, 8-8
and Globalization Support, D-6
querying audit log, 14-8
_ldifwrite
log file location, 14-1
LDIF files
creating, 7-6
for adding configuration set entries, 7-5
importing by using bulkload, 9-4
importing, by using bulkload, 9-4
referencing in commands, 7-7
removing proprietary data from in migration, 27-7
using, 5-8
ldifmigrator, 5-12
ldifwrite, 5-11
Less or Equal filter, A-18, A-31
line-mode commands, batching, 11-7
listener, for directory database, 3-3, 3-4
restarting, 30-10
stopping, 30-10
listener.ora, 30-10
load balancing
and replication, 22-4
-load option, in bulkload, 9-5
location-independence, of directories, 2-2
log files
debug, viewing, 14-5, L-14
locations, 14-1
logging
for garbage collectors, enabling and disabling, 26-7
login
anonymous, A-2
super user, A-2
user, A-2
Index-12

loose consistency model of replication, 22-3
LSNRCTL utility, 30-10

M
managing
directory schema, 11-1
mandatory attributes, 3-12, 11-2
   adding to existing object classes, 11-4
   adding to object classes in use, 8-5
   entering values for, 8-4
   in a user entry, 27-10
   in object classes, A-18, A-19
   redefining, 11-4
manual resolution of conflicts, 30-37
matching rules
   as metadata in schema, 11-1
   attribute, 3-11
   cannot add to subSchemaSubentry, 11-1
   stored in schema, 11-1
Matching Rules tab page, in Oracle Directory Manager, A-22
MD4, 20-3, 27-7
MD5, 20-3, 27-7
   for password encryption, 20-2, 20-4
MD5 Digest, SASL authentication mechanism, 16-4
member attribute, 13-6, 13-9
memory
   in capacity planning, 24-1
   insufficient, 25-5
   required, 22-7
   requirements in capacity planning, 24-9
   tuning, 25-5
   usage, 22-8
   virtual, 24-9
menu bar, Oracle Directory Manager, 5-3
metadata
cache, 3-4
directory, defined, 3-4
   stored in schema, 11-1
metadirectory
deployment considerations, 22-5
middle tier
   using proxy user with, 7-9, 16-5
migrating data, 27-2, 27-6
   from other LDAP directories, 27-2
   from other LDAP-compliant directories, 27-1
migration
   from application-specific repositories, 27-8
      intermediate template file, 27-8
   from other LDAP directories, 27-2
modifiersName attribute, 3-10, 27-7
   optional in top, 3-13
modifyDN, audit log event, 14-11
modifyTimestamp attribute, 3-10, 27-7
   optional in top, 3-13
monitoring servers, 14-13
mpstat utility, 25-2
multimaster flag
toggling, 30-13
multimaster replication, 2-7, 3-17, 22-3, 22-4, 29-4
   groups, 29-5
      in conjunction with fan-out replication
      groups, 29-7
   installation types, 30-2
   installing, 30-6
multiple configuration set entries, 17-3
multiple server processes, 3-4
multithreaded command-line tools
   ldapaddmt, 8-7
multithreaded LDAP servers, 2-7
multivalued attributes, 3-10
   converting to single-valued, 11-9
   member, 13-6, 13-9
   orclEntryLevelACI, 18-3

N
names
   of groups, planning, 23-3
   of users, planning, 23-3
names, of object classes, A-17, A-19
naming contexts, 3-14
   backing up and restoring, 15-1
   definition, 3-14
   discovering, 3-14
   in partitioned directories, 3-18
   managing, 7-8
   publishing, 3-14, 7-8
      by using ldapmodify, 7-8
      by using Oracle Directory Manager, 7-8
   searching for published, 7-8
   subordinate, 3-19
namingContexts attribute, 7-8
   multivalued, 7-8
native authentication
   contrasted with external authentication, 34-1
   defined, 34-1
navigator pane, in Oracle Directory Manager, 5-3
network
   bandwidth, 24-10
   capacity planning, 24-10
   connectivity, in capacity planning, 24-1
   requirements, 24-10
   New Attribute Type Advanced tab page, in Oracle Directory Manager, A-21
   New Attribute Type General tab page, in Oracle Directory Manager, A-21
   New Constraint dialog box, in Oracle Directory Manager, A-5
   New Content Rule dialog box, in Oracle Directory Manager, A-22
new features, i-xxxix
   release 10g (10.1.2), i-xlii
   release 10g (10.1.3), i-xxxix
   release 10g (9.0.4), i-xlili
   release 2.1.1, i-liii
   release 3.0.1, i-li
   release 9.0.2, i-xlviii

Index-12
New Plug-in dialog box, in Oracle Directory Manager, A-9, A-10
New Replica Agreement Naming Context page, in Oracle Directory Manager, A-15
new syntaxes, adding, 3-11
NLS_LANG environment variable, D-3
setting, D-3
in the client environment, D-7
settings, D-3
no SSL authentication option, A-3
nodes, Oracle Internet Directory, 3-2
non-default port, running on, 5-2
not null filter, in Oracle Directory Manager, A-18
null values, in attributes, 11-2
number of worker threads used in change log processing, modifying, 31-3

O
o attribute, 3-10
O3LOGON algorithm, 20-4
object
adding, by using Oracle Directory Manager, 5-4
object class types
structural, 3-13
object classes, 3-12
adding, 11-2
by using command-line tools, 11-7
by using Oracle Directory Manager, 11-6
as metadata in schema, 11-1
assigning to entries, 11-2
auxiliary, 3-13
converting auxiliary, 11-4
creating, by using Oracle Directory Manager, 5-4
defining, 11-16
deleting
by using Oracle Directory Manager, 11-7
from base schema, 11-17
not in base schema, 11-5
explosion, 11-3
extensibleObject, 8-10
groupOfNames, 13-6, 13-7, 13-9, 13-10
guidelines
for adding, 11-3
for deleting, 11-17
for modifying, 11-4
in the base schema, modifying, 11-4
managing
by using command-line tools, 11-7
by using Oracle Directory Manager, 11-2
modifying, 11-4
by using command-line tools, 11-7
by using Oracle Directory Manager, 11-6
orclacpgroup, 18-3
orclauditoc, 14-9
orclprivilegegroup, 3-5
and dynamic groups, 13-3
redefining mandatory attributes in, 11-4
referral, 8-10
removing attributes from, 11-4
removing superclasses from, 11-4
rules, 3-13
searching for, 11-5
searching for, by using Oracle Directory Manager, 11-5
structural, 3-13
structural, converting, 11-4
subclasses, 3-12
defining, 11-16
superclasses, 3-12
top, 3-12
types, 3-12
abstract, 3-13
auxiliary, 3-13
structural, 3-12
types of, 3-12
unique name of, 11-4
unique object identifier, 11-4
viewing, 11-6
viewing properties, 11-6
object identifier as attribute alias, 11-20
object identifiers, of object classes, A-18, A-19
objectclass attribute, 14-9
objects
adding, by using a template, 5-5
adding, by using Oracle Directory Manager, 5-5
comparing, 5-4
modifying
by using ldapmodify, 8-8
by using Oracle Directory Manager, 5-4, 5-5
of ACI directives, 18-6
removing
by using Oracle Directory Manager, 5-4, 5-5
searching for
by using Oracle Directory Manager, 5-4, 5-5
searching for, by using Oracle Directory Manager, 5-5
OID Control Utility
restart command, 7-3
viewing debug log files, 14-5, L-14
OID Database Password Utility, 7-11
OID Database Password Utility (oidpasswd), 5-13
OID Database Statistics Tool, 5-12
OID Migration Tool, 5-12
OID Monitor, 3-3, 5-9
log file location, 14-2
OID Password Utility, 4-3
OID Reconciliation Tool, 5-12, 30-38
oidcmprec
limitations, 31-18
OIDCTL, 6-7
oidctl
viewing debug log files, 14-5, L-14
oidctl. See OID Control Utility
oidexaup.sql
contents of, 34-4
for installing external authentication plug-in, 34-2
OIDEXTAUTH PL/SQL package for external
Index-14

authentication, 34-2
oidldapd
    log file location, 14-2
OIDMON, 6-6
oidstats.sql, 5-12
OLTS_ATTRSTORE tablespace, 24-9
OLTS_CT_STORE tablespace, 24-9
OLTS_DEFAULT tablespace, 24-9
one-level search, 8-2
one-way authentication, SSL, A-3
online administration tool. See Oracle Directory Manager
online directories, 2-1
open cursors parameter, 25-7
OPEN_CURSORS, 25-7
OpenLDAP Community, 0-xxxvii
operation debug dimension, 14-6
operational attributes, 7-7
    ACI, 16-3
Operations menu item, in Oracle Directory Manager, 5-4
operations, limiting debugging to specific, 14-6
OPMN, 6-2
optimizing searches, 25-9
optional attributes, 3-12, 11-2
    adding to pre-defined object classes, 11-16
    entering values for, 8-4
    in object classes, A-18, A-19
options, attribute, 3-11
Oracle, 3-7
Oracle Advanced Database replication architecture, 29-21
Oracle Advanced Security, use of Oracle Internet Directory, 2-8
Oracle Application Server Administrators Group, 21-9
Oracle Application Server Certificate Authority part of Oracle Identity Management, 2-4
Oracle Application Server Portal, use of Oracle Internet Directory, 2-8
Oracle Application Server Single Sign-On use of Oracle Internet Directory, 2-8
Oracle background processes, 25-8
Oracle Call Interface
Oracle Collaboration Suite, use of Oracle Internet Directory, 2-8
Oracle components
    privileges for administering, 21-3
Oracle components, use of Oracle Internet Directory, 2-7
Oracle Context root, 23-9
Oracle Context Administrators Group, 21-13
Oracle data servers
    changing password to, 7-11
    error messages, L-2
Oracle Database Advanced Replication, 29-20, 30-11
    configuring, 30-9, 30-11
    by using Replication Management Tool, 30-9
    for directory replication, 30-11
directory replication based on, 3-17, 29-4
    installing, 30-9
    setting up, 30-9
Oracle Database Advanced replication conflict resolution, 29-25
Oracle Delegated Administration Services overview, 3-26
    part of Oracle Identity Management, 2-4
Oracle Directory Integration Platform, 2-9
    what it is, 3-22, 22-5
Oracle Directory Manager, 8-2
    Access Control Management pane, A-4
    adding
        ACPs, 18-16
        attributes, 11-11
        configuration set entries, 7-3
        entries, 8-3
        group entries, 8-5, 13-6
        object classes, 11-6
        objects, 5-4
    Apply button vs. OK button, 5-3
    ASR Agreement tab page, A-13
    Attributes tab page, A-19
    attributes, searching for, 11-10
    Authentication Choice list, A-4
    By Whom tab page, A-4
    Cancel button, 5-3
    Change Log window, A-17
    Configuration Sets General tab page, A-24
    connecting to a directory server, 5-4, 5-5
    Content Rule dialog box, A-23
    create access control policy point menu, 5-4
    Create button, 5-5
    Create Entry menu item, 5-4
    Create Like button, 5-5, 8-4
    Create Like operation, 5-4
    creating an attribute, 5-4
    creating object classes, 5-4
defined
    Delete button, 5-5
deleting
        configuration set entries, 7-3
        objects, 5-5
disconnecting from a directory server, 5-4
    displaying help navigator, 5-4
    Edit button, 5-5
    Edit menu, 5-4
    Edit Plug-in dialog box, A-10, A-11, A-12
    Encryption Choice list, A-4
    Ends With filter, A-18
    entries management, 5-7
    Exact Match filter, A-18, A-31
    Exit menu item, 5-4
    File menu, 5-4
    Find Attributes button, 11-10
    Find Objects button, 5-5, 11-5
    Garbage Collector window, A-6
    granting access, 18-14
    Greater or Equal filter, A-18, A-31
    Help button, 5-5
Index-14
Help menu item, 5-4
launching, 5-2
Less or Equal filter, A-18, A-31
managing
ACPs, 5-7
configuration set entries, 7-3
entries, 5-7
object classes, 11-2
Matching Rules tab page, A-22
menu bar, 5-3
modifying
configuration set entries, 3-6, 7-3
entries, 8-5
object classes, 11-6
objects, 5-4, 5-5
replication agreements, 31-6
navigating, 5-3
New Attribute Type Advanced tab page, A-21
New Attribute Type General tab page, A-21
New Constraint dialog box, A-5
New Content Rule dialog box, A-22
New Plug-in dialog box, A-9, A-10
New Replica Agreement Naming Context page, A-15
not null filter, A-18
on UNIX, starting, 5-2
on Windows, starting, 5-2
Operations menu, 5-4
overview, 5-1, 5-4
Password Policies Account Lockout tab page, A-7
Password Policies General tab page, A-6
Password Policies IP Lockout tab page, A-8
Password Policies Password Syntax tab page, A-8
Password Verifier Profile dialog box, A-8
Present filter, A-31
Query Optimization tab page, A-29
Refresh button, 5-5
Refresh Entry button, 5-5
Refresh Subtree Entries button, 5-5
removing objects, 5-4
Replica Agreement Replica Naming Context tab page, A-15
Replica Agreement Window, A-16
Replica Agreements tab page, A-14
Replica Node General tab page, A-14
Replication Server Configuration Set General tab page, A-12
Revert button, 5-3
root of search, 8-1
running, 5-2
schema administration, 5-8
search criteria bar, 8-2, 14-13
search filters, 11-5
searching
entries, 8-1
for an object, 5-5
for attributes, 11-10
selecting attribute syntax type, 11-26
Server Chaining Management Window, A-35
SSL Settings tab page, A-32
starting, 5-2
on UNIX, 5-2
on Windows, 5-2
Synchronization Execution tab page, A-34
Synchronization General tab page, A-33
Synchronization Mapping tab page, A-35
Synchronization Status tab page, A-35
system operation attributes displayed, 31-6
Synchronization Execution tab page, A-34
Synchronization General tab page, A-33
Synchronization Mapping tab page, A-35
Synchronization Status tab page, A-35
Systems Passwords tab page, A-29
tear-off menu item, 5-4
toolbar, 5-5
updating, 5-4
subtree entry data, 5-5
View menu, 5-4
viewing attributes, 8-3
Oracle Directory Platform
part of Oracle Identity Management, 2-4
Oracle directory replication server
authentication, 29-18
component of Oracle Internet Directory, 2-6
component of Oracle Internet Directory node, 3-3
collection parameter, location, 31-1
starting, 30-13
uses LDAP to communicate to directory server, 3-3
Oracle directory server instance, 3-3
Oracle directory server instances, 2-6, 3-3
managing, 7-1
starting, 30-13
stopping, 4-1
Oracle directory version field, in Oracle Directory Manager, A-26
Oracle foreground processes
tuning CPU for, 25-4
Oracle Globalization Support, 3-15
Oracle Identity Management, 3-23
and Oracle Internet Directory, 2-4, 23-1
components, 3-24
delegation in, 21-1
group information, 23-4
in application deployments, 2-4
infrastructure, 3-23
what it does, 3-23
management policies, 3-25
objects, 23-9
planning, 23-2
realms, planning, 23-5
user information, 23-4, 23-11
Oracle Internet Directory
advantages of, 2-7
and Oracle Identity Management, 2-4
architecture, 2-5, 3-1
components, 2-6
how Oracle components use it, 2-7
multiple installations on same host, 22-5
nodes, 3-2
used by Oracle Advanced Security, 2-8
used by Oracle Application Server Single Sign-On, 2-8
Oracle Internet Directory Self-Service Console, 3-21
Index-16

in indirect authentication of end users, 16-5
Oracle Internet Directory Server Manageability architecture and components, 14-14
capabilities, 14-14
configuring, 14-16
framework, 14-13
configuring critical events, 14-19
location of configuration information, 14-16
viewing information, 14-19
Oracle Net Services, 3-3, 3-7
preparing for replication, 30-9
use of Oracle Internet Directory, 2-8
Oracle Real Application Clusters, i-lii
Oracle Application Server Single Sign-On part of Oracle Identity Management, 2-4
Oracle database, 3-3
orclACI, 18-2
access to, 18-2
optional attribute in top, 3-13
orclacpgroup object class, 18-3
orclAgreementID, 31-6
orclauditlevel attribute, 14-11
orclauditlevel operational attribute, 14-8
orclauditmessage attribute, 14-9
orclauditobject attributes, 14-9
orclauditobject class, 14-9
orclChangeRetryCount, 31-3
orclcommonusernickname
uniqueness constraint, 10-7
orcldebugflag, 14-5
orclDirReplGroupDSAs, 31-7
orclEntryLevelACI, 18-2, 18-3
optional attribute in top, 3-13
orcleventtime attribute, 14-9
orcleventtype attribute, 14-9
orclexcludedattributes, 29-14
orclexcludednamingcontexts, 29-14
orclGuid
optional attribute in top, 3-13
orclusername attribute, 7-10
orclupassword attribute, 7-10
orclincludednamingcontexts, 29-13
ORCLLM algorithm, 20-4
ORCLMAXCC, 25-3
orclmaxcc, 3-4
ORCLNT algorithm, 20-4
orclopresult attribute, 14-9
orclpkimatchingrule, 16-5
orclprivilegegroup object class, 3-5
and dynamic groups, 13-3
orclprname attribute, 7-10
orclprpassword attribute, 7-10
orclrevpwd attribute, 20-2
orclsequence attribute, 14-9, 14-10
ORCLSERVERPROCS, 25-3
orclskewedattribute attribute, 25-11
orclusername attribute, 7-10
orclsupassword attribute, 7-10
orcluniqueattribname, 10-2
orcluniqueenable, 10-2
orcluniqueobjectclass, 10-2
orcluniquescope, 10-2
orcluniquesubtree, 10-2
orcluserdn attribute, 14-9
orclUserV2 attribute, 27-10
ORCLWEBDAV algorithm, 20-4
organization attribute, 3-10
organizationalUnitName, 3-10
overall throughput, 25-1

P

parameters
configuration, for Oracle directory replication server, 31-1
dependent on Oracle directory server configuration, 25-8
for an active instance, modifying, 17-3
in an active server instance modifying, 7-3
replication agreement, 31-6
required for tuning, 25-7
SGA, 25-8
partial replication, 3-17, 29-2
partitioning, 3-16, 3-18
deployment considerations, 22-3
partitions, 22-2
password policies, 16-7
about, 19-1
categorization, 16-7
creating by using Oracle Directory Manager, 19-9
default, 19-4
definition, 19-1
defined, 3-6
tab page, in Oracle Directory Manager, A-7
tab page, in Oracle Directory Manager, A-6
for realms
modifying by using command-line tools, 19-11
viewing by using command-line tools, 19-10
management, 3-15
managing by using command-line tools, 19-10
modifying by using Oracle Directory Manager, 19-9
plug-in, 33-1
how it works, 33-1
realms, managing by using command-line tools, 19-10
setting by using command-line tools, 19-10
setting by using Oracle Directory Manager, 19-8
troubleshooting, 19-7
verification of, 19-7
viewing by using Oracle Directory Manager, 19-9
Password Policies Account Lockout tab page, in Oracle Directory Manager, A-7
Password Policies General tab page, in Oracle Directory Manager, A-6
Password Policies IP Lockout tab page, in Oracle
Directory Manager, A-8
Password Policies Password Syntax tab page, in Oracle Directory Manager, A-8
password verifier entry, defined, 3-6
Password Verifier Profile dialog box, in Oracle Directory Manager, A-8
password verifiers
default, for Oracle components, 20-7
password wallets
troubleshooting, L-21
password-based authentication, 16-4, A-2
passwords
database, 7-11
for guest users, 7-9
for proxy users, 7-9
for shell tools, 9-4
for SSL wallets, A-3
for super user, 7-9
for super users, 7-9
forcing changes by using command-line tools, 19-12
integrity
MD4, 20-2
policies, 16-7
setting by using command-line tools, 19-10
setting by using Oracle Directory Manager, 19-8
protection, 3-15, 16-7
changing by using ldapmodify, 20-3
changing by using Oracle Directory Manager, 20-3
changing scheme, 20-1
default verifiers for Oracle components, 20-7
managing by using ldapmodify, 20-3
managing by using Oracle Directory Manager, 20-3
MD5, 20-2, 20-4
O3LOGON, 20-4
ORCLLM, 20-4
ORCLNT, 20-4
ORCLWEBDAV, 20-4
SASL/MD5, 20-4
setting by using Oracle Directory Manager, A-27
SHA, 20-2, 20-4
UNIX Crypt, 20-3, 20-4
to a directory, changing, 7-9
to Oracle data servers, changing, 7-11
peer-to-peer replication, 3-17, 29-4
performancadd or modify, L-9
by using orclEntryLevelACI, 18-2
metrics, 25-1
replication and, 22-4
search, L-8
troubleshooting, L-8
tuning, tools for, 25-2
permissions, 3-15, 16-3
granting
by using command-line tools, 18-35
by using Oracle Directory Manager, 18-14
physical distribution, partitions and replicas, 22-2
physical memory, 24-9
PKI authentication, 16-2
plug-ins
adding, 32-5, 32-6
deleting, 32-7
tentry, 3-6
external authentication, 34-1
for password policies, 33-1
framework, 32-1
garbage collection, 26-1
modifying, 32-7
password policy
how it works, 33-1
registering
by using command-line tools, 32-6
by using Oracle Directory Manager, 32-5
point-to-point replication, 3-17, 29-4
policies
identity management, 3-25
pooling, connection, 2-7
port, A-3
default, 5-2
precedence
at the attribute level, 18-12
at the entry level, 18-11
rules
ACL evaluation, 18-11
in conflicting access policies, 18-2
prescriptive access control, 18-2
Present filter, Oracle Directory Manager, A-31
privacy, data, 3-15, 16-1
by using SSL, 2-7
privilege groups, 18-3
associated with orclPrivilegeGroup object class, 18-4
defined, 3-5
privileges, 3-15, 16-3
privileges for user and group management
delegation of, 21-3
process instance location, A-27
processes, 3-3
Oracle background, 25-8
processing power of CPU, 24-11
processor affinity on SMP systems, 25-4
proxy users, 16-5
definition, 7-9
managing, 7-9
by using ldapmodify, 7-10
by using Oracle Directory Manager, 7-9
user name and password, 7-9
public key infrastructure, 16-2
pwdPolicy object class, 19-4
Q
queries, database
optimizing, 25-10
query entry return limit, A-27
Index-18

Query Optimization tab page, in Oracle Directory Manager, A-29
querying
 audit log, 14-8
 critical events, 14-8

R
RC4_40 encryption, 16-2
RDNs. See relative distinguished names (RDNs)
realms, 23-7
 identity management
 configuring, 23-12
 customizing, 23-12
 default, 3-25, 23-10
 defined, 3-25
 implementation in Oracle Internet Directory, 23-9
 in enterprise deployments, 23-7
 in hosted deployments, 23-8
 multiple in enterprise deployments, 23-8
 planning, 23-5
 single in enterprise, 23-7
realm-specific Oracle Context, 23-10
recovery features, in Oracle, 2-7
redefining mandatory attributes, 11-4
redo log buffers parameter, 25-8
redundancy
 and failover, 22-2
 ref attribute, 8-10
 referral caching, client-side, 8-11
 how it works, 8-12
 referral object class, 8-10
 referrals, 3-19
 client-side referral caching, 8-11
 defined, 3-19
 kinds, 3-20
Refresh button, in Oracle Directory Manager, 5-5
Refresh Entry button, in Oracle Directory Manager, 5-5
Refresh Entry menu item, 5-4
Refresh Subtree Entries button, in Oracle Directory Manager, 5-5
Refresh Subtree Entries menu item, 5-4
relational databases contrasted to directories, 2-2
relative distinguished names (RDNs)
 displaying for each entry, 8-1
 modifying
 by using command line tools, 8-8
 modifying, by using ldapmoddn, 8-8
remtool, 30-11
replica
 subentry, 29-9
Replica Agreement Replica Naming Context tab page, in Oracle Directory Manager, A-15
Replica Agreement Window, in Oracle Directory Manager, A-16
Replica Agreements tab page, in Oracle Directory Manager, A-14
Replica Node General tab page, in Oracle Directory Manager, A-14
replica state, H-1
replicas, 3-17, 22-2, 29-3
 in deployment, 22-2
replicated directories, conceptual discussion, 3-16
replication, 3-17, 14-2, 29-25
 adding a new entry to a consumer, F-1
 adding a new node for, 30-14
 agreement entry, 29-10
 agreement parameters, 31-6
 modifying, 31-6, 31-7
 viewing and modifying, 31-6, 31-7
 agreements, 3-17, 29-3, 31-6, A-27
 adding nodes to, 31-7
 configuring, 31-6
 example of, 29-15
 and SSL, 29-19
 architecture, 29-21
 authentication, 29-18
 change conflicts
 monitoring, 30-37
 change logs, 2-7, 29-23
 change logs in, 29-19
 configuration parameters
 modifying, 31-2
 configuring
 Oracle Database Advanced Replication, 30-11
 sqlnet.ora, 30-9
 tnsnames.ora, 30-9
 conflicts
 levels of occurrence, 29-25
 resolving manually, 30-37
 typical causes of, 29-26
 considerations, 22-3
 deleting a node, 30-18
 deleting an entry, F-2
 failure tolerance, 22-4
 fan-out, 3-17, 29-4, 29-6
 full, 3-17, 29-2
 groups, 29-4
 fan-out, 29-6
 multimaster, 3-17, 29-5
 single-master, 29-5
 in deployment, 22-3
 installing and configuring, 30-6
LDAP, 29-23
 filtering, 29-31
 process, 29-23
LDAP-based, 3-17, 29-4
 configuring, 30-20, 30-22
 deleting, 30-33
 determining what is to be replicated, 30-33
 installing and configuring, 30-19
load balancing, 22-4
log location, A-27
login events, 14-10
loose consistency model, 22-3
managing, 30-1
managing naming contexts and attributes, 29-35
modifying a DN, F-4
Index-19

modifying an RDN,  F-3
multimaster,  2-7, 3-17, 22-3, 29-4
    installing and configuring,  30-5
multimaster with fan-out,  29-7
multimaster, single-master, fan-out,  29-6
naming context container entry,  29-13
naming contexts
    included and excluded,  29-30
nodes
    adding,  30-14
    deleting,  30-18
Oracle Advanced Database
    architecture,  29-21
Oracle Database Advanced
    conflict resolution,  29-25
Oracle Database Advanced Replication,  29-4
Oracle Database Advanced Replication-based,  3-17
Oracle Databased Advanced
    filtering,  29-30
overview,  29-1
partial,  3-17, 29-2
    optimization,  29-36
peer-to-peer,  3-17, 29-4
point-to-point,  3-17, 29-4
preparing the Oracle Net Services environment for,  30-9
process,  F-1, F-2, F-3, F-4
reasons to implement,  22-4
retries
    applying changes,  29-25
security,  29-18
server
    log file location,  14-1
    single-master,  3-17
    specifying number of worker threads,  A-13
status location,  A-27
Replication Environment Management Tool,  5-12
Replication Management Tool,  30-9
Replication Server Configuration Set General tab page, in Oracle Directory Manager,  A-12
resource access information,  3-26
resource information,  3-26
    location in DIT,  3-26
resource type information,  3-26
Revert button, in Oracle Directory Manager,  5-3
Root DSE entry
    defined,  3-5
root of search
    entering,  8-1
    selecting,  8-2
root Oracle Context,  23-9

S

SASL
    clients enabled with
        Digest-MD5 authentication to directory server,  16-8
        external authentication,  16-8
SASL/MD5, for generating password verifier,  20-4
scalability, of Oracle Internet Directory,  2-7
schema
    adding and changing object classes (online),  11-2
    administration,  11-1
        by using Oracle Directory Manager,  5-8
    definition location,  A-27
    definitions in subSchemaSubentry,  11-1
    directory, defined,  3-4
elements
    add/replace event,  14-10
    delete event,  14-10
    for orclACI, C-1
    for orclEntryLevelACI,  C-2
objects, administering by using Oracle Directory Manager,  5-8
scripts, batched line-mode commands,  11-7
search
    and compare operations,  3-11
    criteria bar, in Oracle Directory Manager,  8-2, 14-13
depth, specifying,  8-2
    results, specifying maximum number of entries returned,  8-2, 14-12
Search ACPs
    button,  5-5
    menu item,  5-4
searches
    configuring,  7-11
        for ACPs when using Oracle Directory Manager,  18-15
configuring display and duration of,  5-6
duration,  14-13
    specifying maximum number of entries returned,  8-2, 14-12
    using filters,  11-5
secure
    port 636,  17-2, 17-3
    Secure Hash Algorithm (SHA),  20-3, A-27
secure mode
    running server instances in,  17-3
security,  2-7, 3-14
    credentials, stored in an external repository,  34-1
    for different clients,  17-3
    in LDAP Version 3,  2-4
    in replication,  29-18
    SSL parameters for different clients,  17-3
    within Oracle Internet Directory environment,  3-15
Security Administrators Group,  21-12
security and refresh events garbage collector,  26-2
selected audit log events,  14-11
server
    instances
        running,  5-2
        running in secure mode,  17-3
    mode,  A-27
    operation time limit,  A-27
Server Chaining Management Window, in Oracle Directory Manager,  A-35
subentries, definition, 11-1
subordinate naming contexts, 3-19
subSchemaSubentry
holding schema definitions, 11-1
subtree entry data, updating by using Oracle Directory Manager, 5-5
subtree level search, 8-2
subtrees
displaying, 8-1
super users
definition, 7-9
logging in as, A-2
login events, 14-10
managing, 7-9
by using ldapmodify, 7-10
by using Oracle Directory Manager, 7-9
user name and password, 7-9
superclass selector, 8-3
superclasses, 3-12
and inheritance, 11-2
of object classes, A-18, A-19
superior knowledge references (referrals), 3-19
suppliers
defined, 3-17, 29-3
surname attribute, 3-10
Symmetric Multi-Processor (SMP) systems, 25-4
Synchronization Execution tab page, in Oracle Directory Manager, A-34
Synchronization General tab page, in Oracle Directory Manager, A-33
Synchronization Mapping tab page, in Oracle Directory Manager, A-35
Synchronization Status tab page, in Oracle Directory Manager, A-35
syntax
attribute, 3-11
stored in schema, 11-1
syntaxes
cannot add to subSchemaSubentry, 11-1
new, adding, 3-11
viewing
by using by using ldapsearch, 11-26
by using Oracle Directory Manager, 11-26
System Global Area (SGA), 25-5
parameters, 25-8
sizing, 25-5
tuning for Oracle, 25-5
tuning parameters, 25-8
system operation attributes
displayed in Oracle Directory Manager, A-24
system operational attributes, 7-7
setting, 7-7
by using ldapmodify, 7-7
by using Oracle Directory Manager, 7-7
viewing, 7-7
System Passwords tab page, in Oracle Directory Manager, A-29
system resource events garbage collector, 26-3
SYSTEM tablespace, 24-9
T
tables, 24-6
OLTS_ATTRSTORE, 24-9
OLTS_CT_STORE, 24-9
OLTS_DEFAULT, 24-9
sizing, 24-6
SYSTEM, 24-9
TCP/IP Problems, L-1
tear-off, in Oracle Directory Manager, 5-4
templates, creating entries from, 8-4
throughput, 24-5
overall, 25-1
time-based change log purging, 26-5
tnsnames.ora
configuring for replication, 30-9
tombstone garbage collector, 26-3
tools
for tuning, 25-2
top object class, 3-12, 3-13
optional attributes in, 3-13
top utility, 25-2
tracing function calls, 14-6
tree view
browsing, 8-2
selecting root of search, 8-2
troubleshooting, L-1
change log garbage collection, L-20
directory server instance startup, L-9
dynamic password verifiers, L-21
password policies, L-7
password wallets, L-21
performance, L-8
SSL setup, L-20
typical problems in Oracle Internet Directory, L-1
Trusted Application Administrators Group, 21-10
tunables, database, 25-7
tuning, 22-5, 25-1
considerations, 22-8
CPU for Oracle foreground processes, 25-4
CPU for Oracle Internet Directory processes, 25-3
CPU usage, 25-2
deployment considerations, 22-8
disk, 25-6
memory, 25-5
overview, 25-1
SGA parameters, 25-8
System Global Area (SGA) for Oracle, 25-5
tools, 25-2

types
of attributes, 3-9
of object classes, A-18, A-19
U
Unicode Transformation Format 8-bit (UTF-8)
uniqueness constraint
orclcommonusernickname, 10-7
UNIX crypt
for password hashing, 20-3
UNIX, starting Oracle Directory Manager on, 5-2
unspecified access, 18-10, 18-23
user
  login, A-2
  names and passwords, managing
    by using ldapmodify, 7-10
    by using Oracle Directory Manager, 7-9
  password modification event, 14-10
user certificates, searching directory for, G-1
User field, in Oracle Directory Manager, A-2
User Management Application Administrators Group, 21-10
User Preferences
  button, 5-5
  menu item, 5-4
User Proxy Privilege Group, 21-13
userCertificate attribute, G-1
userPassword attribute, hash values, 27-7
users
  entries
    adding by using ldapadd, 8-8
    adding by using Oracle Directory Manager, 8-4
    modifying by using ldapmodify, 8-8
    modifying by using Oracle Directory Manager, 8-5
  guest, 7-9
  names and containment, planning, 23-3
  proxy, 7-9, 16-5
  super, 7-9
UTF-8. See Unicode Transformation Format 8-bit
UTLBSTAT.SQL, 25-2
UTLESTAT.SQL, 25-2

V
Verifier Services Group, 21-12
View menu, in Oracle Directory Manager, 5-4
virtual memory, 24-9
vmstat utility, 25-2

W
wallets
  passwords, A-3
wildcards, in setting access control policies, 18-37
Windows
  starting Oracle Directory Manager on, 5-2
  Task Manager, 25-2
Windows Performance Monitor, 25-2
worker threads, 25-8
  specifying in replication, A-13