Oracle® Identity Governance

Configuring the Database User Management Application

12c (12.2.1.3.0)

F12367-09

May 2021

Oracle Identity Governance Configuring the Database User Management Application, 12c (12.2.1.3.0)

F12367-09

Primary Author: Alakrita.Prakash

Contributing Authors: Gowri.G.R

Contents

Preface

What's New In This Guide

5 Performing the Postconfiguration Tasks

6 Using the Database User Management Connector

List of Figures

List of Tables

Preface

This guide describes the connector that is used to onboard applications pertaining to database user management tables into Oracle Identity Governance.

Audience

This guide is intended for resource administrators and target system integration teams.

Documentation Accessibility

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

Access to Oracle Support

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

Related Documents

For information about installing and using Oracle Identity Governance 12.2.1.3.0, visit the following Oracle Help Center page:

`https://docs.oracle.com/en/middleware/idm/identity-governance/12.2.1.3/index.html`

For information about installing and using Oracle Identity Manager 11.1.2.3, visit the following Oracle Help Center page:

`http://docs.oracle.com/cd/E52734_01/index.html`

For information about Oracle Identity Governance Connectors 12.2.1.3.0 documentation, visit the following Oracle Help Center page:

`https://docs.oracle.com/en/middleware/idm/identity-governance-connectors/12.2.1.3/index.html`

For information about Oracle Identity Manager Connectors 11.1.1 documentation, visit the following Oracle Help Center page:

`http://docs.oracle.com/cd/E22999_01/index.htm`

Conventions

The following text conventions are used in this document:

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

What's New In This Guide

These are the updates made to the software and documentation for release 12.2.1.3.0 of Configuring the Database User Management Application.

The updates provided in this chapter are divided into the following categories:

- Software Updates

  These include updates made to the connector software.

- Documentation-Specific Updates

  These include major changes made to the connector documentation. These changes are not related to software updates.

Software Updates

These are the updates made to the connector software.

Software Updates in Release 12.2.1.3.0

The following is the software update in release 12.2.1.3.0:

Support for Onboarding Applications Using the Connector

From this release onward, the connector bundle includes application onboarding templates required for performing connector operations on the Oracle Database and MySQL targets. This helps in quicker onboarding of the applications for these targets into Oracle Identity Governance by using an intuitive UI.

Documentation-Specific Updates

These are the updates made to the connector documentation.

Documentation-Specific Updates in Release 12.2.1.3.0

The following documentation-specific update has been been made in revision "09" of this guide:

The "Target Systems" row of Table 1-1 has been updated to include support for Microsoft SQL Server 2016.

The following documentation-specific update has been been made in revision "08" of this guide:

The "Target Systems" row of Table 1-1 has been updated to include support for Oracle Database 19c.

The following documentation-specific update has been been made in revision "05" of this guide:

Information about Oracle Identity Manager versions prior to 11g Release 2 PS3 (11.1.2.3.0) has been removed from the guide.

The following documentation-specific updates have been been made in revision "04" of this guide:

The "Target systems" row of Table 1-1 has been updated.

The following documentation-specific updates have been been made in revision "03" of this guide:

Description for parameter **Connection URL** in Table 3-1 has been updated.

The following documentation-specific updates have been been made in revision "02" of this guide:

- The "Oracle Identity Governance or Oracle Identity Manager" row of Table 1-1 has been updated to include support for Oracle Identity Governance release 12*c* PS4 (12.2.1.4.0).
- Several broken links were fixed throughout the document.

The following documentation-specific update has been made in revision "01" of this guide:

This is the first release of the Oracle Identity Governance Connector for Database User Management. Therefore, there are no documentation-specific updates in this release.

1 About the Database User Management Connector

The Database User Management connector integrates Oracle Identity Governance with database user management tables in Oracle Database, Microsoft SQL Server, MySQL, IBM DB2, and Sybase.
The following sections provide a high-level overview of the connector:

1.1 Introduction to the Database User Management Connector

Oracle Identity Governance is a centralized identity management solution that provides self service, compliance, provisioning and password management services for applications residing on-premise or on the Cloud. Oracle Identity Governance connectors are used to integrate Oracle identity Governance with the external identity-aware applications.

The Database User Management connector lets you onboard applications in Oracle Identity Governance for target systems such as Oracle Database and MySQL.

> **Note:**
>
> In this guide, the connector that is deployed using the **Applications** option on the **Manage** tab of Identity Self Service is referred to as an **AOB application**. The connector that is deployed using the **Manage Connector** option in Oracle Identity System Administration is referred to as a **CI-based connector** (Connector Installer-based connector).

From Oracle Identity Governance release 12.2.1.3.0 onward, connector deployment is handled using the application onboarding capability of Oracle Identity Self Service. This capability lets business users to onboard applications with minimum details and effort. The connector installation package includes a collection of predefined templates (XML files) that contain all the information required for provisioning and reconciling data from a given application or target system. These templates also include basic connectivity and configuration details specific to your target system. The connector uses information from these predefined templates allowing you to onboard your applications quickly and easily using only a single and simplified UI.

Application onboarding is the process of registering or associating an application with Oracle Identity Governance and making that application available for provisioning and reconciliation of user information.

You can configure and use the Database User Management connector with the following target systems:

- Oracle Database

  In Oracle Database, the Login and User entities are treated as a single entity. In this guide, that entity is referred to as the Login entity.

- MySQL

> **Note:**
>
> In this guide, database resources such as Oracle and MySQL are referred to as the **target system.**

1.2 Certified Components

These are the software components and their versions required for installing and using the connector.

> **Note:**
>
> If you are using Oracle Identity Manager release 11.1.*x*, then you can install and use the connector only in the CI-based mode. If you want to use the AOB application, then you must upgrade to Oracle Identity Governance release 12.2.1.3.0 or later.

Table 1-1 Certified Components

Component	Requirement for AOB Application	Requirement for CI-Based Connector
Oracle Identity Governance or Oracle Identity Manager	You can use one of the following releases: • Oracle Identity Governance 12c (12.2.1.4.0) • Oracle Identity Governance 12c (12.2.1.3.0)	You can use one of the following releases: • Oracle Identity Governance 12c (12.2.1.4.0) • Oracle Identity Governance 12c (12.2.1.3.0) **Note:** If you are using Oracle Identity Governance 12c (12.2.1.3.0), then download and apply the patch 26616250 from My Oracle Support. Failing to apply this patch causes target resource user reconciliation runs to fail. • Oracle Identity Manager 11*g* Release 2 PS3 (11.1.2.3.0)

Table 1-1 (Cont.) Certified Components

Component	Requirement for AOB Application	Requirement for CI-Based Connector
Target systems	The target system can be any one of the following: • Oracle Database 12*c* as single database, pluggable database (PDB), or Oracle RAC implementation • Oracle Database 18c as single database, pluggable database (PDB), or Oracle RAC implementation • Oracle Database 19c as single database, pluggable database (PDB), or Oracle RAC implementation • MySQL 5.*x*	The target system can be any one of the following: • Exadata V2 • Oracle Database 12*c* as single database, pluggable database (PDB), or Oracle RAC implementation • Oracle Database 18c as single database, pluggable database (PDB), or Oracle RAC implementation • Oracle Database 19c as single database, pluggable database (PDB), or Oracle RAC implementation • Microsoft SQL Server 2005, 2008, 2012, 2014, 2016 • MySQL 5.*x* • IBM DB2 UDB 9.*x* • Sybase 15.*x*
Connector Server	12.2.1.3.0	12.2.1.3.0
Connector Server JDK	JDK 1.8 or later.	JDK 1.8 or later.

1.3 Usage Recommendation

These are the recommendations for the Database User Management connector versions that you can deploy and use depending on the Oracle Identity Governance or Oracle Identity Manager version that you are using.

- If you are using Oracle Identity Governance 12c (12.2.1.3.0) and want to integrate it with Oracle Database or MySQL, then use the latest 12.2.1.x version of this connector and deploy it using the **Applications** option on the Manage tab of Identity Self Service.
- If you are using Oracle Identity Governance 12c (12.2.1.3.0) and want to integrate it with IBM DB2, Microsoft SQL Server, or Sybase, then use the latest 12.2.1.x version of this connector and deploy it using the **Manage Connector** option in Oracle Identity System Administration.
- If you are using any of the Oracle Identity Manager releases listed in the “Requirement for CI-Based Connector” column of Certified Components, then use the 11.1.1.*x* version of the Database User Management connector. If you want to use the 12.1.*x* version of this connector, then you can install and use it only in the CI-based mode. If you want to use the AOB application, then you must upgrade to Oracle Identity Governance release 12.2.1.3.0.

- If you are using a release earlier than Oracle Identity Manager 11*g* Release 1 (11.1.1.5.3) and no later than Oracle Identity Manager release 9.1.0.2, then use the 9.1.*x* version of the Database User Management connector.

1.4 Certified Languages

These are the languages that the connector supports.

- Arabic
- Chinese (Simplified)
- Chinese (Traditional)
- Czech
- Danish
- Dutch
- English
- Finnish
- French
- French (Canadian)
- German
- Greek
- Hebrew
- Hungarian
- Italian
- Japanese
- Korean
- Norwegian
- Polish
- Portuguese
- Portuguese (Brazilian)
- Romanian
- Russian
- Slovak
- Spanish
- Swedish
- Thai
- Turkish

1.5 Supported Connector Operations

These are the list of operations that the connector supports for your target system.

Table 1-2 Supported Connector Operations

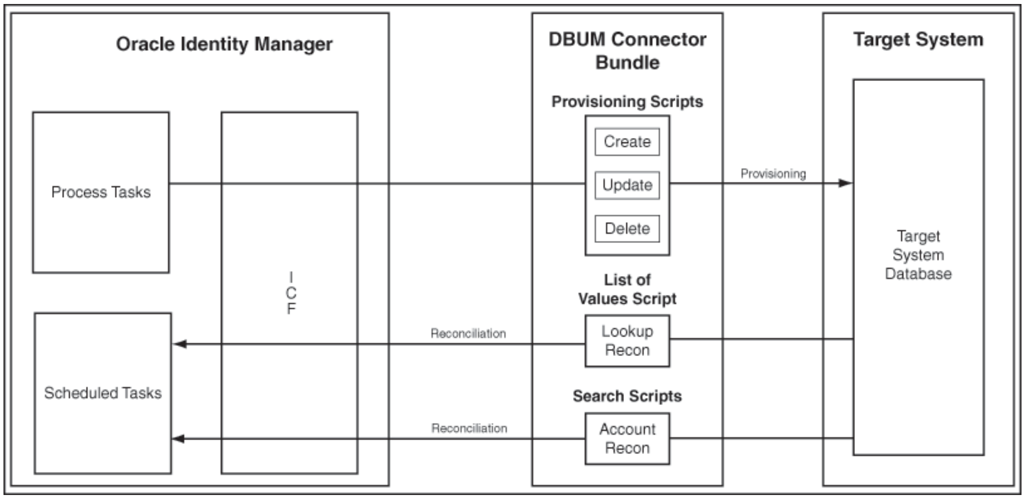
Operation	Supported for IBM DB2?	Supported for MSSQL?	Supported for MySQL?	Supported for Oracle Database ?	Supported for Sybase?
User Management					
Create user	Yes	Yes	Yes	Yes	Yes
Update user	No	No	No	Yes	Yes
Delete User	Yes	Yes	Yes	Yes	Yes
Enable user	Yes	No	No	Yes	Yes
Disable user	Yes	No	No	Yes	Yes
Reset password	Yes	Yes	Yes	Yes	Yes
Create UserLogin	Not applicable	Yes	Not applicable	Not applicable	Yes
Update UserLogin	Not applicable	Yes	Not applicable	Not applicable	Yes
Delete UserLogin	Not applicable	Yes	Not applicable	Not applicable	Yes
Entitlement Grant Management					
Add roles	Not applicable	Yes	No	Yes	Not applicable
Revoke Roles	Not applicable	Yes	No	Yes	Not applicable
Add privileges	Not applicable	Not applicable	Yes	Yes	Not applicable
Revoke privileges	Not applicable	Not applicable	Yes	Yes	Not applicable
Add schema	Yes	Not applicable	Not applicable	Not applicable	Not applicable
Revoke schema	Yes	Not applicable	Not applicable	Not applicable	Not applicable
Add tablespace	Yes	Not applicable	Not applicable	Not applicable	Not applicable
Revoke tablespace	Yes	Not applicable	Not applicable	Not applicable	Not applicable
Add roles list	Not applicable	Not applicable	Not applicable	Not applicable	Yes
Revoke roles list	Not applicable	Not applicable	Not applicable	Not applicable	Yes

1.6 Connector Architecture

The Database User Management connector enables management of database accounts through Oracle Identity Governance, and is implemented using the Identity Connector Framework (ICF).

Figure 1-1 shows the architecture of the connector.

Figure 1-1 Architecture of the Connector



The Database User Management connector is implemented by using the Identity Connector Framework (ICF). The ICF is a component that provides basic reconciliation and provisioning operations that are common to all Oracle Identity Governance connectors. In addition, ICF provides common features that developers would otherwise need to implement on their own, such as connection pooling, buffering, time outs, and filtering. The ICF is shipped along with Oracle Identity Governance. Therefore, you need not configure or modify the ICF.

> **See Also:**
>
> Understanding the Identity Connector Framework in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance* for more information about the ICF

The out of the box (OOB) connector is provided with scripts for the certified targets Oracle, MSSQL, MySQL, DB2, and Sybase. If you customize the connector for a database other than the certified ones, then you need to manually add scripts for the new database.

The connector performs all DBUM operations by executing SQL Scripts or by calling Stored Procedures (Procs).

For example, during a provisioning operation, the process tasks invoke an ICF operation, ICF inturn invokes an operation on the connector bundle which runs corresponding SQL statements. These SQL statements carry out the required operation on the target system, and return the response from the target system back to the bundle, which passes it to the adapters.

Similarly, during reconciliation, a scheduled task invokes ICF operation, ICF inturn invokes a search operation on the connector bundle which runs the corresponding query or stored procedure on the target system. Target system records that meet the query or stored procedure criteria are fetched into Oracle Identity Governance.

The scripts and stored procs in the connector bundle are externalized in different files in the connector bundle and can also be customized. The bundle key is made of bundle name, bundle version, connector name, and is used for loading the bundle.

The following are the three categories of scripts that are stored in the connector bundle:

Script	Description
Provisioning.queries	This script is used for Create, Update, or Delete operations.
LoVSearch.queries	This script is used for lookup reconciliation. It contains the set of possible values for certain fields such as profile, privileges, roles, and tablespaces.
Search.queries	This script is used for full or incremental or delete reconciliation. You can also perform account and group search with various conditions using this script.

Depending on the query invoked, ExecutionHandler executes the queries. There are two different handlers SQLExecutionHandler and StoredProcExecutionHandler which extends ExecutionHandler.

Depending on the type of Query, the corresponding ExecutionHandler is invoked. StoredProcExecutionHandler is used for operations in MSSQL. The following is an example used for searching users:

```
USER_DATA_QUERY {
    Query="CALL sp_helpuser({__UID__})"
    QueryType="StoredProc"
    Parameters=["__UID__":"Type:String,Direction:IN",
            "defaultDatabase":"Type:String,Direction:OUT,ColName:DefDBName",
            "loginName":"Type:String,Direction:OUT,ColName:LoginName",

"roles~DBRole~__NAME__":"Type:String,Direction:OUT,ColName:RoleName"]
    QueryExtensions=[]
}
```

SQL queries are categorized into Data Definition Language (DDL) and Data Manipulation Language (DML) queries. DDL queries are used for CREATE, REVOKE, GRANT, ALTER, and so on, where as DML queries are used for UPDATE, INSERT, and so on.

The DDL queries are executed as regular statements. The following is an example for the DDL statement used for the create operation:

```
Statement stmt = null;
                try {
                        stmt = _dbConnection.getConnection().createStatement();
                        stmt.execute(sqlScript);
                }
```

The DML queries are executed as prepared statements. The following is an example for the DML statement used for the update operation:

```
PreparedStatement st = null;
            try {
                st = conn.prepareStatement(sqlScript);
                setParams(st, Arrays.asList(params));
                return st.executeUpdate();
                }
```

The information about the connector bundle is stored in the manifest file. This file contains the connector definition, which gives the information about the connector bundle framework version, connector bundle name, and connector bundle version. The following is the example of the connector definition which is required to identity a connector bundle:

org.identityconnectors.dbum.12.3.0.jar

In this example:

org.identityconnectors: refers to connector bundle framework

dbum: refers to the connector bundle name

12.3.0 jar: refers to the connector bundle version

1.7 Supported Connector Features Matrix

Provides the list of features supported by the AOB application and CI-based connector.

Table 1-3 Supported Connector Features Matrix

Feature	AOB Application	CI-Based Connector	Supported Target Systems
Add new standard and custom attributes for reconciliation and provisioning	Yes	Yes	All
Customize the predefined queries for reconciliation	Yes	Yes	All
Customize the predefined queries for provisioning	Yes	Yes	All
Full reconciliation	Yes	Yes	All
Incremental reconciliation	Yes	Yes	• **For AOB Application:** Oracle Database • **For CI-based connector:** Oracle Database and MSSQL
Limited reconciliation	Yes	Yes	All
Batched reconciliation	Yes	Yes	All
Exclude accounts from reconciliation and provisioning operations	Yes	Yes	All

Table 1-3 (Cont.) Supported Connector Features Matrix

Feature	AOB Application	CI-Based Connector	Supported Target Systems
Connection pooling	Yes	Yes	All
Use connector server	Yes	Yes	All
Clone applications or create new application instances	Yes	Yes	All
Transformation and validation of account data	Yes	Yes	All
Reconcile deleted entities	Yes	Yes	All
Scheduled jobs for reconciliation of users, logins, and deleted login entities	Yes	Yes	All
SSL communication between the target system and Oracle Identity Manager	Yes	Yes	All
Add pre or post action scripts	Yes	Yes	All
Manage authorizations to Oracle Database Vault realms	Yes	Yes	Oracle Database
Configure Enterprise User Security	Yes	Yes	Oracle Database

1.8 Connector Features

The features of the connector include support for connector server, predefined and custom queries for performing provisioning and reconciliation operations, reconciliation of all existing or modified account data, support for limited and batched reconciliation, and so on.

The following are features of the connector:

- Mapping Standard and Custom Attributes for Reconciliation and Provisioning
- Predefined and Custom Provisioning and Reconciliation Queries
- Full and Incremental Reconciliation
- Limited (Filtered) Reconciliation
- Batched Reconciliation
- Specifying Accounts to Be Excluded from Reconciliation and Provisioning Operations
- Connection Pooling
- Support for Connector Server
- Support for Cloning Applications and Creating Instance Applications

- Transformation and Validation of Account Data
- Support for Reconciling Data About Deleted Entities
- Separate Scheduled Jobs for Reconciliation of Users, Logins, and Deleted Login Entities
- Support for SSL Communication Between the Target System and Oracle Identity Governance
- Support for Running Pre and Post Action Scripts
- Support for Managing Authorization to Oracle Database Vault Realms
- Support for Configuring the Connector for Enterprise User Security

1.8.1 Mapping Standard and Custom Attributes for Reconciliation and Provisioning

You can create mappings for single-valued and multivalued target system attributes that are not included in the list of default attribute mappings. These attributes can be part of the standard set of attributes provided by the target system or custom attributes that you add on the target system.

For more information about adding new attributes, see Providing Schema Information for Target Application or Providing Schema Information for Authoritative Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

1.8.2 Predefined and Custom Provisioning and Reconciliation Queries

Provisioning involves running a SQL query or stored procedure such as CREATE USER, ALTER USER, and DROP USER to perform Create User and Update user operations on the target system through Oracle Identity Governance. Reconciliation involves running a SQL query or stored procedure on the target system database to fetch the required user account records to Oracle Identity Governance. The connector provides predefined SQL queries and stored procedures that enable you to reconcile user data from the target system and perform provisioning operations such as create, enable, and update target system accounts. You can modify these SQL queries or stored procedures. In addition, you can add your own SQL queries or stored procedures for provisioning and reconciliation.
The predefined SQL queries and stored procedures for reconciliation and provisioning are stored in the Search.queries and Provisioning.queries files, respectively, in the connector bundle.

For more information about modifying predefined SQL queries and stored procedures, see Modifying the Predefined Queries or Creating New Queries.

1.8.3 Full and Incremental Reconciliation

Full reconciliation involves reconciling all existing user records from the target system into Oracle Identity Governance. In incremental reconciliation, only records that are added or modified after the last reconciliation run are fetched into Oracle Identity Governance.

After you create the application, you can first perform full reconciliation. After the first full reconciliation run, incremental reconciliation is automatically enabled.

The following sections provide more information:

- Performing Full and Incremental Reconciliation from Oracle Database
- Performing Full Reconciliation from MySQL

1.8.4 Limited (Filtered) Reconciliation

ICF filter performs the limited reconciliation and the records are fetched into Oracle Identity Governance during a reconciliation run. The ICF filters are translated to WHERE clause and applied in the Search query.

The following sections provide more information:

- Performing Limited Reconciliation from Oracle Database
- Performing Limited Reconciliation from MySQL

1.8.5 Batched Reconciliation

You can break down a reconciliation run into batches by specifying the number of records that must be included in each batch and the query that must be used to perform batched reconciliation.

The following sections provide more information:

- Performing Batched Reconciliation from Oracle Database
- Performing Batched Reconciliation from MySQL

1.8.6 Specifying Accounts to Be Excluded from Reconciliation and Provisioning Operations

You can specify a list of target system accounts that must be excluded from all reconciliation and provisioning operations. The accounts for which you specify users attributes in the exclusion list are not affected by reconciliation and provisioning operations.

You can write a Groovy-based validation script that specifies a list of accounts that must be excluded from connector operations. For more information about the Validation Groovy Script for Resource Exclusion, see About Customizing Groovy Scripts in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

1.8.7 Connection Pooling

A connection pool is a cache of objects that represent physical connections to the target. Oracle Identity Governance connectors can use these connections to communicate with target systems.

At run time, the application requests a connection from the pool. If a connection is available, then the connector uses it and then returns it to the pool. A connection returned to the pool can again be requested for and used by the connector for another operation. By enabling the reuse of connections, the connection pool helps reduce connection creation overheads like network latency, memory allocation, and authentication.

One connection pool is created for each set of basic configuration parameters that you provide while creating an application. For example, if you have three applications for three installations of the target system, then three connection pools will be created, one for each target system installation.

For more information about configuring connection pool, see:

- For Oracle Database: Advanced Settings Parameters for Oracle Database
- For MySQL: Advanced Settings Parameters for MySQL

1.8.8 Support for Connector Server

Connector Server is a component provided by ICF, and it enables remote execution of an Oracle Identity Governance connector. By using one or more connector servers, the connector architecture permits your application to communicate with externally deployed bundles.

A Java connector server is useful when you do not wish to execute a Java connector bundle in the same VM as your application. It can be beneficial to run a Java connector on a different host for performance improvements.

For information about installing, configuring, and running the Connector Server, and then installing the connector in a Connector Server, see Using an Identity Connector Server in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*.

1.8.9 Support for Cloning Applications and Creating Instance Applications

You can configure this connector for multiple installations of the target system by cloning applications or by creating instance applications.

When you clone an application, all the configurations of the base application are copied into the cloned application. When you create an instance application, it shares all configurations as the base application.

For more information about these configurations, see Cloning Applications and Creating Instance Applications in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

1.8.10 Transformation and Validation of Account Data

You can configure transformation and validation of account data that is brought into or sent from Oracle Identity Governance during reconciliation and provisioning operations by writing Groovy scripts while creating your application.

For more information, see Validation and Transformation of Provisioning and Reconciliation Attributes in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

1.8.11 Support for Reconciling Data About Deleted Entities

You can reconcile data about login entities that have been deleted on the target system that has been configured as a trusted source or target resource.

After the records are fetched into Oracle Identity Governance, depending on whether you have configured your target system as a target resource or trusted source, the records are compared with existing OIM Users or database resources provisioned to existing OIM Users. The unmatched accounts are revoked/removed from Oracle Identity Governance.

1.8.12 Separate Scheduled Jobs for Reconciliation of Users, Logins, and Deleted Login Entities

You can reconcile data about users, logins, or deleted login entities from a target system that is configured as a trusted source or target resource. Depending on the target system that you are using, the mode in which it is configured, and the type of data that you want to reconcile, separate scheduled jobs have been created.

For information about the scheduled jobs, see one of the following topics:

- Reconciliation Jobs for Oracle Database
- Reconciliation Jobs for MySQL

1.8.13 Support for SSL Communication Between the Target System and Oracle Identity Governance

You can configure SSL to secure communication between Oracle Identity Governance and the target system.

The following sections provide more information:

- Configuring Secure Communication Between Oracle Database and Oracle Identity Governance
- Configuring Secure Communication Between MySQL and Oracle Identity Governance

1.8.14 Support for Running Pre and Post Action Scripts

You can run pre and post action scripts on a computer where the connector is deployed. These scripts can be of type SQL/StoredProc/Groovy. You can configure the scripts to run before or after the create, update, or delete an account provisioning operations.

For more information, see Updating the Provisioning Configuration in *Oracle® Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

1.8.15 Support for Managing Authorization to Oracle Database Vault Realms

Oracle Database Vault restricts access to specific areas in an Oracle Database from any user, including users who have administrative access. For example, you can restrict administrative access to employee salaries, customer medical records, or other sensitive information.

This enables you to apply fine-grained access control to your sensitive data in a variety of ways. It hardens your Oracle Database instance and enforces industry standard best practices in terms of separating duties from users with administrative access. Most importantly, it protects data from super-privileged users but still allows them to manage the Oracle Database installation.

With Oracle Database Vault, you can address business requirements such as protecting against insider threats, meeting regulatory compliance requirements, and enforcing separation of duty.

You configure Oracle Database Vault to manage the security of an individual Oracle Database instance. You can install Oracle Database Vault on standalone Oracle Database installations, in multiple Oracle homes, and in Oracle Real Application Clusters (Oracle RAC) environments.

In Oracle Database installations on which Oracle Database Vault is installed, the connector can be used to grant and manage authorization to Oracle Database Vault realms. The connector treats access to Oracle Database Vault realms as an entitlement. You can use the connector to provision database users with access to multiple realms with different levels of access.

Because Oracle Identity Governance is an enterprise application for managing user accounts and access to entitlements, the connector does not support management of the following:

- Realms
- Command rules and rule sets
- Factors
- Secure Application Roles

See Creating the Administrator Account on Oracle Database Vault for more information.

1.8.16 Support for Configuring the Connector for Enterprise User Security

Oracle Enterprise User Security addresses user, administrative, and security challenges by using the identity management services supplied by Oracle Internet Directory, an LDAP-compliant directory service.

You must use either Oracle Identity Manager LDAP connectors or some other means to create the user in the LDAP-compliant directory. Enterprise users are provisioned and managed centrally in an LDAP-compliant directory, such as Oracle Internet Directory, for database access. Enterprise users have a unique identity in the directory called the distinguished name (DN). When enterprise users log on to a database, the database authenticates those users by using their DN.

In Oracle Database installations configured with Oracle Enterprise User Security, the connector supports the creation of password, and globally authenticated users for a target system account (login or user). Depending on the authentication type, you need to select the corresponding authentication type at the time of provisioning. If the authentication type is global, then you must make the following changes in the process form:

Remove the password field as it is not required for global authentication.

In addition, while performing the provisioning operation, you must:

- Set authentication type to `Global`.
- Provide the unique ID in Global DN.

You can use the connector to create and manage accounts of these enterprise users on the target database.

2
Creating an Application By Using the Database User Management Connector

Learn about onboarding applications using the connector and the prerequisites for doing so.

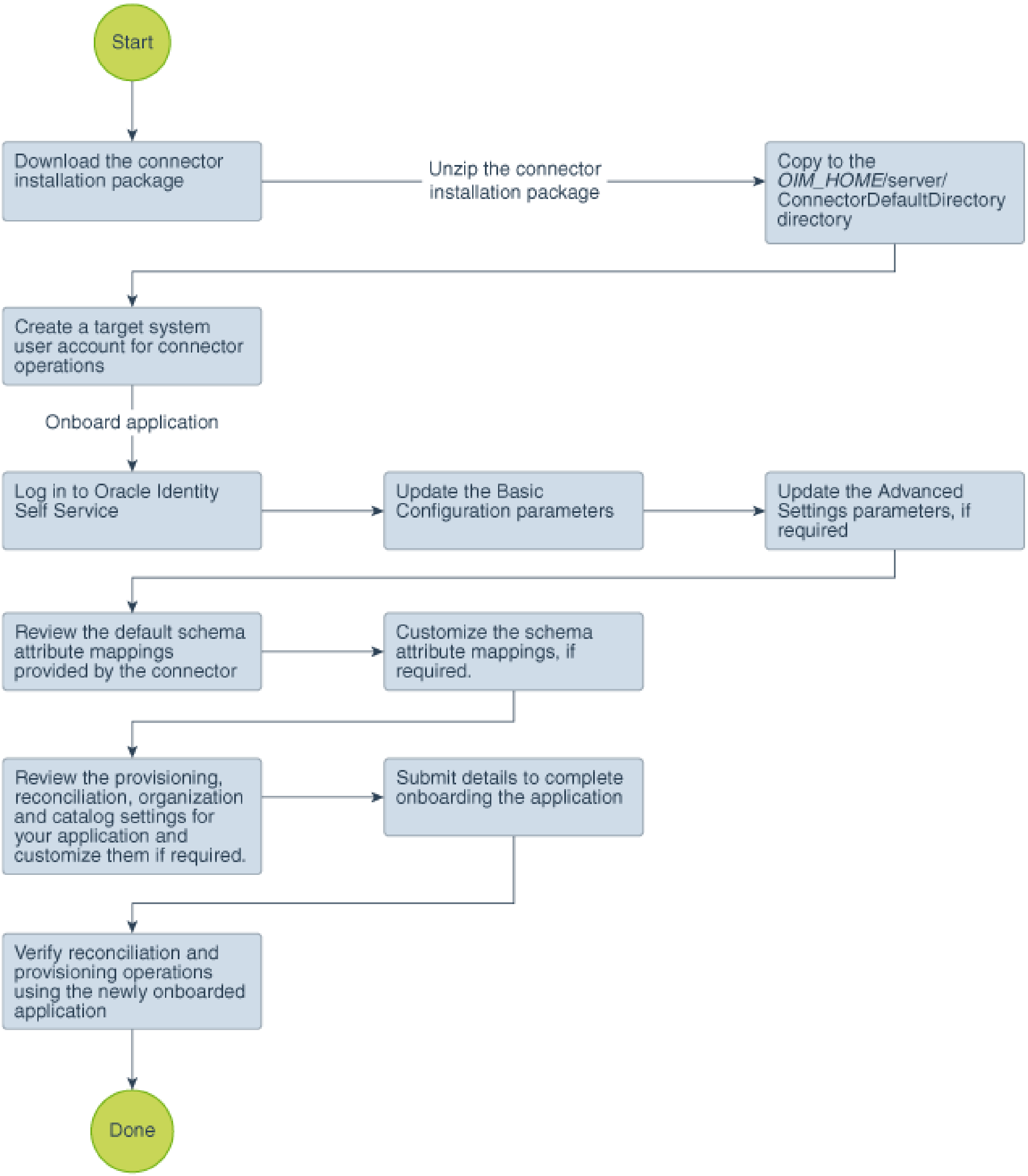
- Process Flow for Creating an Application By Using the Connector
- Prerequisites for Creating an Application Database User Management Connector
- Creating an Application By Using the Connector

2.1 Process Flow for Creating an Application By Using the Connector

From Oracle Identity Governance release 12.2.1.3.0 onward, connector deployment is handled using the application onboarding capability of Identity Self Service.

Figure 2-1 is a flowchart depicting high-level steps for creating an application in Oracle Identity Governance by using the connector installation package.

Figure 2-1 Overall Flow of the Process for Creating an Application By Using the Connector


Start
Download the connector installation package
Unzip the connector installation package
Copy to the *OIM_HOME*/server/ ConnectorDefaultDirectory directory
Create a target system user account for connector operations
Onboard application
Log in to Oracle Identity Self Service
Update the Basic Configuration parameters
Update the Advanced Settings parameters, if required
Review the default schema attribute mappings provided by the connector
Customize the schema attribute mappings, if required.
Review the provisioning, reconciliation, organization and catalog settings for your application and customize them if required.
Submit details to complete onboarding the application
Verify reconciliation and provisioning operations using the newly onboarded application
Done

2.2 Prerequisites for Creating an Application Database User Management Connector

Learn about the tasks that you must complete before you create the application.

- Downloading the Connector Installation Package
- Copying Third-Party JAR Files
- Creating a Target System User Account for Database User Management Connector Operations

2.2.1 Downloading the Connector Installation Package

You can obtain the installation package for your connector on the Oracle Technology Network (OTN) website.

To download the connector installation package:

1. Navigate to the OTN website at http://www.oracle.com/technetwork/middleware/id-mgmt/downloads/connectors-101674.html.
2. Click **OTN License Agreement** and read the license agreement.
3. Select the **Accept License Agreement** option.

   You must accept the license agreement before you can download the installation package.
4. Download and save the installation package to any directory on the computer hosting Oracle Identity Governance.
5. Extract the contents of the installation package to any directory on the computer hosting Oracle Identity Governance. This creates a directory named *CONNECTOR_NAME-RELEASE_NUMBER*.
6. Copy the *CONNECTOR_NAME-RELEASE_NUMBER* directory to the *OIG_HOME*/server/ConnectorDefaultDirectory directory.

2.2.2 Creating a Target System User Account for Database User Management Connector Operations

Oracle Identity Governance requires a target system user account to access the target system during reconciliation and provisioning operations. Depending on the target system you are using, you can create the user in your target system and assign specific permissions and roles to the user.

You provide the credentials of this user account as part of Basic Configuration Parameters for Oracle Database or Basic Configuration Parameters for MySQL while creating an application.

> **See Also:**
>
> Target system documentation for detailed information about creating the user

- For Oracle Database:
  1. Create Login using the following query:

     ```
     CREATE USER serviceuser IDENTIFIED BY password
     DEFAULT TABLESPACE users
     TEMPORARY TABLESPACE temp QUOTA UNLIMITED ON users;
     ```

  2. Assign the following permissions and roles to the created user:
     - `GRANT CONNECT TO serviceuser;`
     - `GRANT SELECT on dba_role_privs TO serviceuser;`
     - `GRANT SELECT on dba_sys_privs TO serviceuser;`
     - `GRANT SELECT on dba_ts_quotas TO serviceuser;`
     - `GRANT SELECT on dba_tablespaces TO serviceuser;`
     - `GRANT SELECT on dba_users TO serviceuser;`
     - `GRANT CREATE USER TO serviceuser;`
     - `GRANT ALTER ANY TABLE TO serviceuser;`
     - `GRANT GRANT ANY PRIVILEGE TO serviceuser;`
     - `GRANT GRANT ANY ROLE TO serviceuser;`
     - `GRANT DROP USER TO serviceuser;`
     - `GRANT SELECT on dba_roles TO serviceuser;`
     - `GRANT SELECT ON dba_profiles TO serviceuser;`
     - `GRANT ALTER USER TO serviceuser;`
     - `GRANT CREATE ANY TABLE TO serviceuser;`
     - `GRANT DROP ANY TABLE TO serviceuser;`
     - `GRANT CREATE ANY PROCEDURE TO serviceuser;`
     - `GRANT DROP ANY PROCEDURE TO serviceuser;`
- For MySQL:
  1. Create a user using the following query:

     ```
     CREATE USER serviceuser IDENTIFIED BY 'password';
     ```

  2. Assign the following permissions and roles to the created user using the following query:

     ```
     GRANT SELECT, INSERT, UPDATE, DELETE, CREATE, ALTER ON *.* TO
     'serviceuser';
     ```

2.2.3 Copying Third-Party JAR Files

These are the drivers that the connector requires to establish a connection with the target system.

- If you are using Oracle database as the target system, then there is no need to copy any JAR files.

- If you are using MySQL as the target system, then copy the mysql-connector-java-5.1.20-bin.jar file to the /ConnectorDefaultDirectory/targetsystems-lib/DBUM-*RELEASE_NUMBER.* directory.

2.3 Creating an Application By Using the Connector

You can onboard an application into Oracle Identity Governance from the connector package by creating a Target application or Authoritative applictaion. To do so, you must log in to Identity Self Service and then choose the **Applications** box on the **Manage** tab.

The following is the high-level procedure to create an application by using the connector:

> **Note:**
>
> For detailed information on each of the steps in this procedure, see Creating Applications of *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

1. Create an application in Identity Self Service. The high-level steps are as follows:
   a. Log in to Identity Self Service either by using the **System Administration** account or an account with the **ApplicationInstanceAdministrator** admin role.
   b. Ensure that the **Connector Package** option is selected when creating an application.
   c. Update the basic configuration parameters to include connectivity-related information.
   d. If required, update the advanced setting parameters to update configuration entries related to connector operations.
   e. Review the default user account attribute mappings. If required, add new attributes or you can edit or delete existing attributes.
   f. Review the provisioning, reconciliation, organization, and catalog settings for your application and customize them if required. For example, you can customize the default correlation rules for your application if required.
   g. Review the details of the application and click **Finish** to submit the application details.

      The application is created in Oracle Identity Governance.
   h. When you are prompted whether you want to create a default request form, click **Yes** or **No**.

      If you click **Yes**, then the default form is automatically created and is attached with the newly created application. The default form is created with the same name as the application. The default form cannot be modified later. Therefore, if you want to customize it, click **No** to manually create a new form and attach it with your application.
2. Verify reconciliation and provisioning operations on the newly created application.

See Also:

- Configuring the Database User Management Connector for Oracle Database or Configuring the Database User Management Connector for MySQL for details on basic configuration and advanced settings parameters, default user account attribute mappings, default correlation rules, and reconciliation jobs that are predefined for this connector
- Configuring Oracle Identity Governance for details on creating a new form and associating it with your application, if you chose not to create the default form.

3
Configuring the Database User Management Connector for Oracle Database

While creating an application, you must configure connection-related parameters that the connector uses to connect Oracle Identity Governance with your target system and perform connector operations. In addition, you can view and edit attribute mappings between the process form fields in Oracle Identity Governance and target system columns, predefined correlation rules, situations and responses, and reconciliation jobs.

3.1 Basic Configuration Parameters for Oracle Database

These are the connection-related parameters that Oracle Identity Governance requires to connect to Oracle Database. These parameters are common for both target applications and authoritative applications.

Table 3-1 Parameters in the Basic Configuration Section for Oracle Database

Parameter	Mandatory ?	Description
Connection URL	Yes	Enter the database connection string using the `host:post:sid` or `host:port/service` syntax format. Default value: `jdbc:oracle:thin:@%h:%p:%s`
User	Yes	Enter the user name of the target system account to be used for connector operations. Sample value: `sys as sysdba` **Note:** If you are configuring the connector for Oracle Database Vault, then enter the user name of the account you created in Creating the Administrator Account on Oracle Database Vault.
Password	Yes	Enter the password for the user name of the target system account to be used for connector operations. **Note:** If you are configuring the connector for Oracle Database Vault, then enter the password of the account you created in Creating the Administrator Account on Oracle Database Vault.
Database Type	Yes	This parameter identifies the database type (such as Oracle or MySQL) and is used for loading respective scripts.

Table 3-1 (Cont.) Parameters in the Basic Configuration Section for Oracle Database

Parameter	Mandatory ?	Description
Connector Server Name	No	If you created an IT resource of the type “Connector Server”, then enter its name.
Database Drivers	No	Name of the JDBC driver class. Default value: `oracle.jdbc.driver.OracleDriver`
Connection Properties	No	Enter the connection properties for the target system database.

3.2 Advanced Settings Parameters for Oracle Database

These are the configuration-related entries that the connector uses during reconciliation and provisioning operations.

> **Note:**
>
> Unless specified, the parameters in the table are applicable to both target and authoritative applications.

Table 3-2 Advanced Setting Parameters for Oracle Database

Parameter	Mandatory ?	Description
Connector Name	Yes	This parameter holds the name of the connector class. **Default value:** `org.identityconnectors.dbum.DBUMConnector`
Connector Package Name	Yes	This parameter holds the name of the connector bundle package. **Default value:** `org.identityconnectors.dbum`
Connector Package Version	Yes	This parameter hods the version of the connector bundle class. **Default value:** 1.0.1116
disableValuesSet	No	Enter the possible values for the disabled status of a user. **Default value:** `"EXPIRED & LOCKED","LOCKED","EXPIRED"`
Reserve Keywords	No	Enter the list of words that are reserved and are not allowed to be used in the names of the connector artifacts **Default value:** `"DROP","INSERT","ALTER","CREATE","DELETE","UPDATE","GRANT","TRUNCATE","EXEC","TEMPORARY","TABLESPACE","DEFAULT","QUOTA","PROFILE","IDENTIFIED","EXTERNALLY","AS","GLOBALLY","REVOKE","ACCOUNT","UNLOCK","LOCK","CASCADE"` **Note:** This parameter is available only when you are creating a target application.

Table 3-2 (Cont.) Advanced Setting Parameters for Oracle Database

Parameter	Mandatory ?	Description
Unsupported Character Set	No	Enter the characters that are not allowed to be used in the names of the connector artifacts **Default value:** "&","--","~","`","\"" **Note:** This parameter is available only when you are creating a target application.
Pool Max Idle	No	Maximum number of idle objects in a pool. **Sample value:** 10
Pool Max Size	No	Maximum number of connections that the pool can create. **Sample value:** 10
Pool Max Wait	No	Maximum time, in milliseconds, the pool must wait for a free object to make itself available to be consumed for an operation. **Sample value:** 150000
Pool Min Evict Idle Time	No	Minimum time, in milliseconds, the connector must wait before evicting an idle object. **Sample value:** 120000
Pool Min Idle	No	Minimum number of idle objects in a pool. **Sample value:** 1

3.3 Attribute Mappings for Oracle Database

The attribute mappings on the Schema page vary depending on whether you are creating a target application or an authoritative application.

- Attribute Mappings for an Oracle Database Target Application
- Attribute Mappings for an Oracle Database Authoritative Application

3.3.1 Attribute Mappings for an Oracle Database Target Application

The Schema page for a target application displays the default schema (provided by the connector) that maps Oracle Identity Governance attributes to target system columns. The connector uses these mappings during reconciliation and provisioning operations.

Oracle Database User Account Attributes

Table 3-3 lists the user-specific attribute mappings between the process form fields in Oracle Identity Governance and Oracle Database columns. The table also lists whether a specific attribute is used during provisioning or reconciliation and whether it is a matching key field for fetching records during reconciliation.

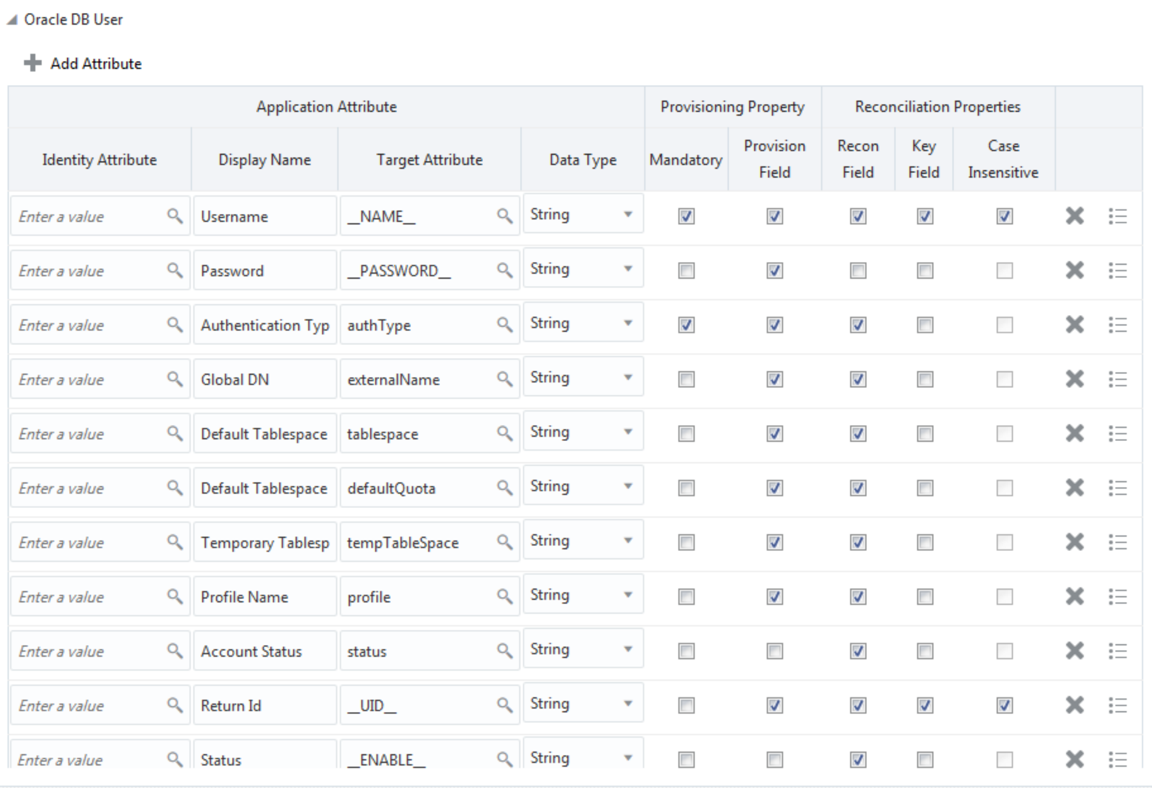
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application of *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-3 Default Attribute Mappings for Oracle DB User Account

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Provision Field?	Recon Field?	Key Field?	Case Insensitive?
Username	__NAME__	String	Yes	Yes	Yes	Yes	Yes
Authentication Type	authType	String	Yes	Yes	Yes	No	Not applicable
Global DN	externalName	String	No	Yes	Yes	No	Not applicable
Default Tablespace	tablespace	String	No	Yes	Yes	No	Not applicable
Default Tablespace Quota	defaultQuota	String	No	Yes	Yes	No	Not applicable
Temporary Tablespace	tempTableSpace	String	No	Yes	Yes	No	Not applicable
Profile Name	profile	String	No	Yes	Yes	No	Not applicable
Return Id	__UID__	String	No	Yes	Yes	Yes	Yes
Account Status	status	String	No	No	Yes	No	Not applicable
Status	__ENABLE__	String	No	No	Yes	No	Not applicable
Password	__PASSWORD__	String	No	Yes	No	No	Not applicable

Figure 3-1 shows the default User account attribute mappings.

Figure 3-1 Default Attribute Mappings for Oracle Database User Account



Oracle DB User

Application Attribute				Provisioning Property		Reconciliation Properties			
Identity Attribute	Display Name	Target Attribute	Data Type	Mandatory	Provision Field	Recon Field	Key Field	Case Insensitive	
Enter a value	Username	__NAME__	String	☑	☑	☑	☑	☑	✖
Enter a value	Password	__PASSWORD__	String	□	☑	□	□	□	✖
Enter a value	Authentication Typ	authType	String	☑	☑	☑	□	□	✖
Enter a value	Global DN	externalName	String	□	☑	☑	□	□	✖
Enter a value	Default Tablespace	tablespace	String	□	☑	☑	□	□	✖
Enter a value	Default Tablespace	defaultQuota	String	□	☑	☑	□	□	✖
Enter a value	Temporary Tablesp	tempTableSpace	String	□	☑	☑	□	□	✖
Enter a value	Profile Name	profile	String	□	☑	☑	□	□	✖
Enter a value	Account Status	status	String	□	□	☑	□	□	✖
Enter a value	Return Id	__UID__	String	□	☑	☑	☑	☑	✖
Enter a value	Status	__ENABLE__	String	□	□	☑	□	□	✖

Role List Entitlement Attributes

Table 3-4 lists the roles-specific attribute mappings between the process form fields in Oracle Identity Governance and Oracle Database columns. The table lists whether a given attribute is mandatory during provisioning. It also lists whether a given attribute is used during reconciliation and whether it is a matching key field for fetching records during reconciliation.

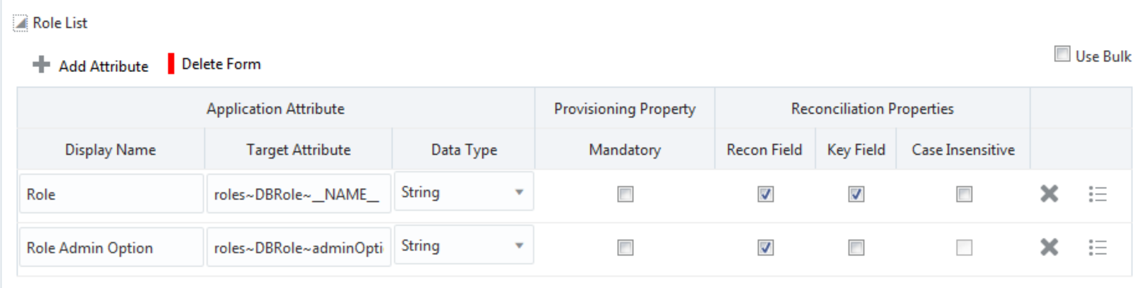
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application of *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-4 Default Attribute Mappings for Oracle Database Role List Entitlement

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Recon Field?	Key Field?	Case Insensitive ?
Role	roles~DBRole~__NAME__	String	No	Yes	Yes	No
Role Admin Option	roles~DBRole~adminOption	String	No	Yes	No	Not applicable

Figure 3-2 shows the default Role List entitlement mapping.

Figure 3-2 Default Attribute Mappings for Oracle Database Role List Entitlement


Role List

Application Attribute			Provisioning Property	Reconciliation Properties			
Display Name	Target Attribute	Data Type	Mandatory	Recon Field	Key Field	Case Insensitive	
Role	roles~DBRole~__NAME__	String	□	☑	☑	□	✖
Role Admin Option	roles~DBRole~adminOpti	String	□	☑	□	□	✖

Privilege List Entitlement Attributes

Table 3-5 lists the roles-specific attribute mappings between the process form fields in Oracle Identity Governance and Oracle Database columns. The table lists whether a given attribute is mandatory during provisioning. It also lists whether a given attribute is used during reconciliation and whether it is a matching key field for fetching records during reconciliation.

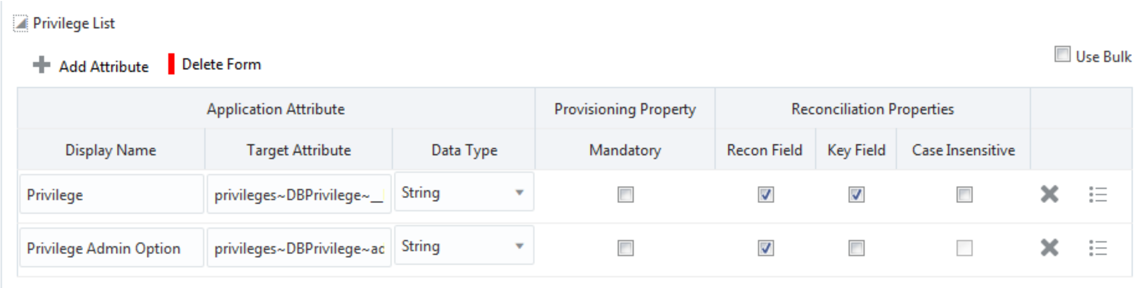
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application of *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-5 Default Attribute Mappings for Oracle Database Privilege List Entitlement

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Recon Field?	Key Field?	Case Insensitive?
Privilege	privileges~DBPrivilege~__NAME__	String	No	Yes	Yes	No
Privilege Admin Option	privileges~DBPrivilege~adminOption	String	No	Yes	No	Not applicable

Figure 3-2 shows the default Privilege List entitlement mapping.

Figure 3-3 Default Attribute Mappings for Oracle Database Privilege List Entitlement


Privilege List

Application Attribute			Provisioning Property	Reconciliation Properties			
Display Name	Target Attribute	Data Type	Mandatory	Recon Field	Key Field	Case Insensitive	
Privilege	privileges~DBPrivilege~_	String	□	☑	☑	□	✖
Privilege Admin Option	privileges~DBPrivilege~ac	String	□	☑	□	□	✖

3.3.2 Attribute Mappings for an Oracle Database Authoritative Application

The Schema page for an authoritative application displays the default schema (provided by the connector) that maps Oracle Identity Governance attributes to target system columns. The connector uses these mappings during reconciliation operations.

Table 3-6 lists the user-specific attribute mappings between the reconciliation fields in Oracle Identity Governance and Oracle Database columns. The table also lists the data type for a given attribute and specified whether it is a mandatory attribute for reconciliation.

If required, you can edit these attributes mappings by adding new attributes or deleting existing attributes on the Schema page as described in Creating an Authoritative Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

You may use the default schema that has been set for you or update and change it before continuing to the next step.

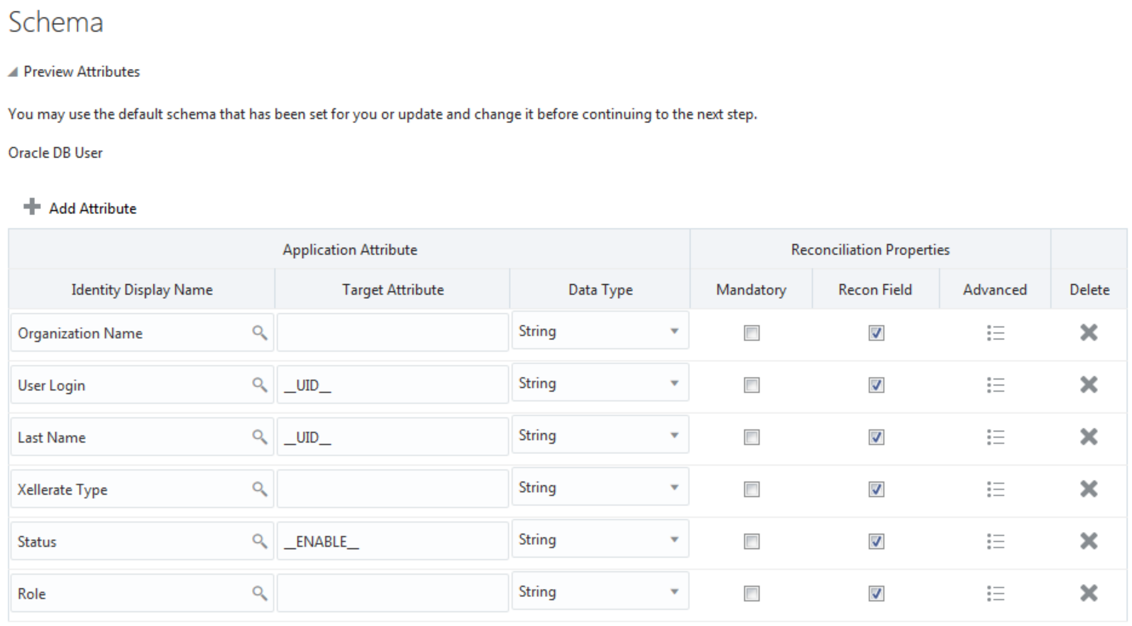
The Organization Name, Xellerate Type, and Role identity attributes are mandatory fields on the OIG User form. They cannot be left blank during reconciliation. The target attribute mappings for these identity attributes are empty by default because there are no corresponding columns in the target system. Therefore, the connector provides default values (as listed in the “Default Value for Identity Display Name” column of Table 3-6) that it can use during reconciliation. For example, the default target attribute value for the Organization Name attribute is Xellerate Users. This implies that the connector reconciles all target system user accounts into the Xellerate Users organization in Oracle Identity Governance. Similarly, the default attribute value for Xellerate Type attribute is End-User, which implies that all reconciled user records are marked as end users.

Table 3-6 Oracle DB User Account Schema Attributes

Identity Display Name	Target Attribute	Data Type	Mandatory Reconciliation Property?	Recon Field?	Default Value for Identity Display Name
Organization Name	NA	String	No	Yes	Xellerate Users
User Login	__UID__	String	No	Yes	NA
Last Name	__UID__	String	No	Yes	NA
Xellerate Type	NA	String	No	Yes	End-User
Status	__ENABLE__	String	No	Yes	NA
Role	NA	String	No	Yes	Full-Time

Figure 3-4 shows the default User account attribute mappings.

Figure 3-4 Default Attribute Mappings for an Oracle Database User Account in an Authoritative Application



Schema

◢ Preview Attributes

You may use the default schema that has been set for you or update and change it before continuing to the next step.

Oracle DB User

Application Attribute			Reconciliation Properties			
Identity Display Name	Target Attribute	Data Type	Mandatory	Recon Field	Advanced	Delete
Organization Name		String	□	☑	☰	✖
User Login	__UID__	String	□	☑	☰	✖
Last Name	__UID__	String	□	☑	☰	✖
Xellerate Type		String	□	☑	☰	✖
Status	__ENABLE__	String	□	☑	☰	✖
Role		String	□	☑	☰	✖

3.4 Correlation Rules for Oracle Database

Learn about the predefined rules, responses and situations for Target and Authoritative applications. The connector use these rules and responses for performing reconciliation.

- Correlation Rules for an Oracle Database Target Application
- Correlation Rules for an Oracle Database Authoritative Application

3.4.1 Correlation Rules for an Oracle Database Target Application

When you create a Target application, the connector uses correlation rules to determine the identity to which Oracle Identity Governance must assign a resource.

Predefined Identity Correlation Rules

By default, the Database User Management connector provides a simple correlation rule when you create a Target application. The connector uses this correlation rule to compare the entries in Oracle Identity Governance repository and the target system repository, determine the difference between the two repositories, and apply the latest changes to Oracle Identity Governance.

Table 3-7 lists the default simple correlation rule for Oracle Database. If required, you can edit the default correlation rule or add new rules. You can create complex correlation rules also. For more information about adding or editing simple or complex correlation rules, see Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-7   Predefined Identity Correlation Rule for an Oracle Database Target Application

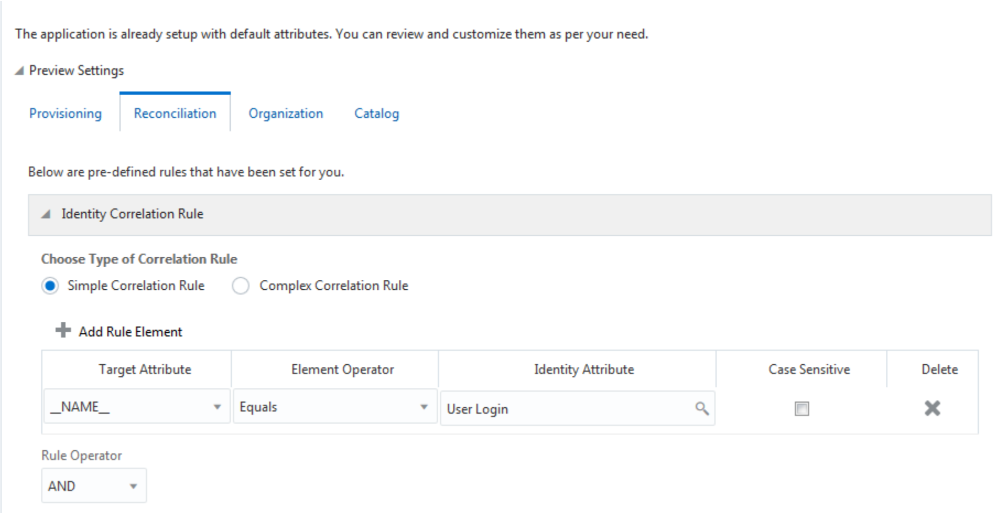
Target Attribute	Element Operator	Identity Attribute	Case Sensitive?
__NAME__	Equals	User Login	No

In this identity rule:

- __NAME__ is a single-valued attribute on the target system that identifies the user account.
- User Login is the field on the OIM User form.

Figure 3-5 shows the simple correlation rule for Oracle Database.

Figure 3-5   Simple Correlation Rule for an Oracle Database Target Application



The application is already setup with default attributes. You can review and customize them as per your need.

Below are pre-defined rules that have been set for you.

Choose Type of Correlation Rule

● Simple Correlation Rule ○ Complex Correlation Rule

Target Attribute	Element Operator	Identity Attribute	Case Sensitive	Delete
NAME	Equals	User Login	□	✖

Rule Operator

AND

Predefined Situations and Responses

The Database User Management connector provides a default set of situations and responses when you create a Target application. These situations and responses specify the action that Oracle Identity Governance must take based on the result of a reconciliation event.

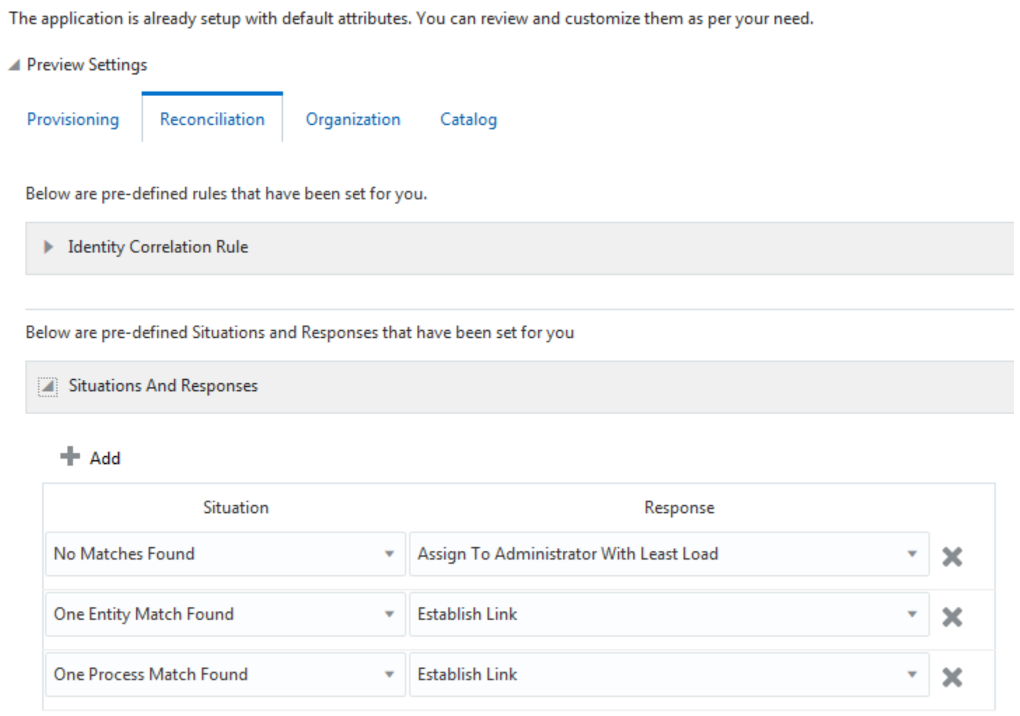
Table 3-8 lists the default situations and responses for Oracle Database. If required, you can edit these default situations and responses or add new ones. For more information about adding or editing situations and responses, see Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*

Table 3-8 Predefined Situations and Responses for an Oracle Database Target Application

Situation	Response
No Matches Found	Assign to Administrator With Least Load
One Entity Match Found	Establish Link
One Process Match Found	Establish Link

Figure 3-6 shows the situations and responses for Oracle Database that the connector provides by default.

Figure 3-6 Predefined Situations and Responses for an Oracle Database Target Application



The application is already setup with default attributes. You can review and customize them as per your need.

◢ Preview Settings

Below are pre-defined rules that have been set for you.

▸ Identity Correlation Rule

Below are pre-defined Situations and Responses that have been set for you

◢ Situations And Responses

Situation	Response	
No Matches Found ▾	Assign To Administrator With Least Load ▾	✖
One Entity Match Found ▾	Establish Link ▾	✖
One Process Match Found ▾	Establish Link ▾	✖

3.4.2 Correlation Rules for an Oracle Database Authoritative Application

When you create an Authoritative application, the connector uses correlation rules to determine the identity that must be reconciled into Oracle Identity Governance.

Predefined Identity Correlation Rules

By default, the Database User Management connector provides a simple correlation rule when you create an Authoritative application. The connector uses this correlation rule to compare the entries in Oracle Identity Governance repository and the target system repository, determine the difference between the two repositories, and apply the latest changes to Oracle Identity Governance.

Table 3-9 lists the default simple correlation rule for an Oracle Database authoritative application. If required, you can edit the default correlation rule or add new rules. You can create complex correlation rules also. For more information about adding or editing simple or complex correlation rules, see Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-9 Predefined Identity Correlation Rule for an Oracle Database Authoritative Application

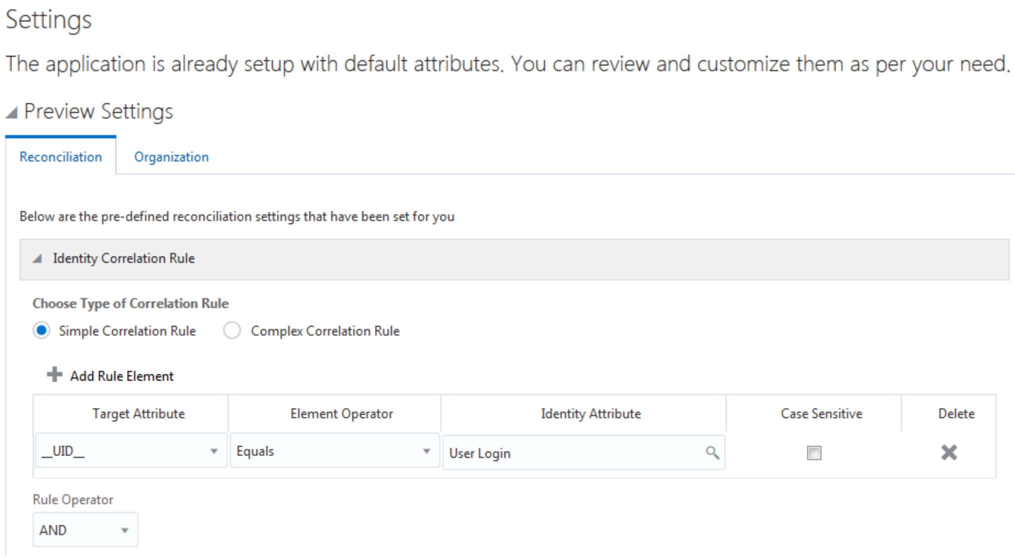
Target Attribute	Element Operator	Identity Attribute	Case Sensitive?
__UID__	Equals	User Login	No

In this identity rule:

- __UID__ is an attribute on the target system that uniquely identifies the user account.
- User Login is the field on the OIM User form.

Figure 3-7 shows the simple correlation rule for an Oracle Database Authoritative application.

Figure 3-7 Simple Correlation Rule for an Oracle Database Authoritative Application



Settings

The application is already setup with default attributes. You can review and customize them as per your need.

Below are the pre-defined reconciliation settings that have been set for you

Choose Type of Correlation Rule

● Simple Correlation Rule ○ Complex Correlation Rule

Target Attribute	Element Operator	Identity Attribute	Case Sensitive	Delete
__UID__	Equals	User Login	□	✖

Rule Operator: AND

Predefined Situations and Responses

The Database User Management connector provides a default set of situations and responses when you create an Authoritative application. These situations and responses specify the action that Oracle Identity Governance must take based on the result of a reconciliation event.

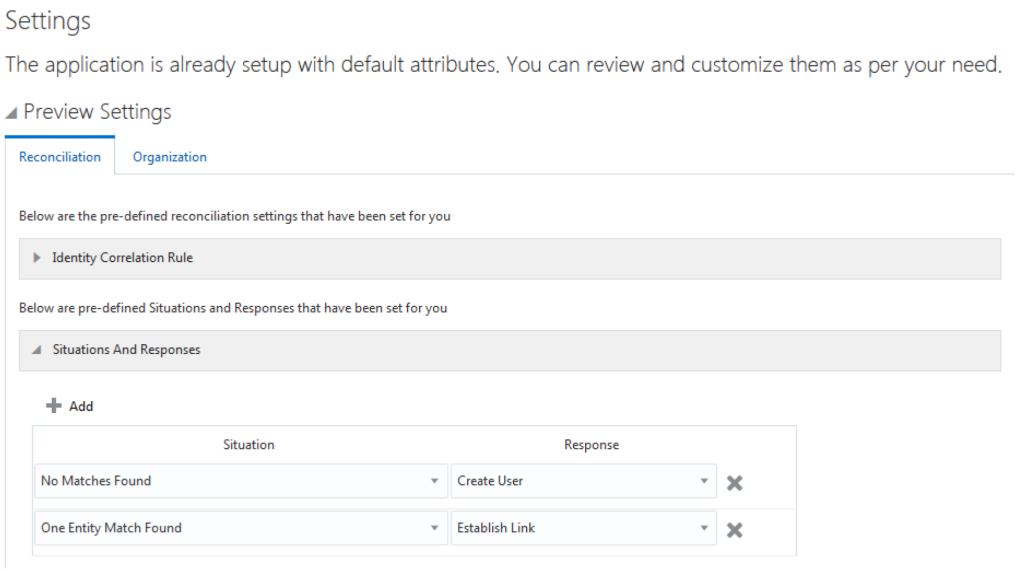
Table 3-10 lists the default situations and responses for Oracle Database. If required, you can edit these default situations and responses or add new ones. For more information about adding or editing situations and responses, see Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*

Table 3-10 Predefined Situations and Responses for an Oracle Database Authoritative Application

Situation	Response
No Matches Found	Create User
One Entity Match Found	Establish Link

Figure 3-8 shows the situations and responses for an Oracle Database Authoritative application that the connector provides by default.

Figure 3-8 Predefined Situations and Responses for an Oracle Database Authoritative Application



Settings

The application is already setup with default attributes. You can review and customize them as per your need.

◢ Preview Settings

Below are the pre-defined reconciliation settings that have been set for you

▸ Identity Correlation Rule

Below are pre-defined Situations and Responses that have been set for you

◢ Situations And Responses

Situation	Response
No Matches Found	Create User
One Entity Match Found	Establish Link

3.5 Reconciliation Jobs for Oracle Database

These are the reconciliation jobs that are automatically created in Oracle Identity Governance after you create the application for your target system.

You can either use these predefined jobs or edit them to meet your requirements. Alternatively, you can create custom reconciliation jobs. For information about editing these predefined jobs or creating new ones, see Updating Reconciliation Jobs in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

User Reconciliation Jobs

The following reconciliation jobs are available for reconciling user data:

- DBUM Oracle User Target Reconciliation: Use this reconciliation job to reconcile user data from a Target application.

- DBUM Oracle User Trusted Reconciliation: Use this reconciliation job to reconcile user data from an Authoritative application.

The parameters for both these jobs are the same.

Table 3-11 Parameters of the User Reconciliation Jobs for Oracle Database

Parameter	Description
Application Name	Name of the application you created for your target system. This value is the same as the value that you provided for the Application Name field while creating your target application. Do *not* modify this value.
Batch Size	Enter the number of records that must be included in each batch fetched from the target system during reconciliation.
Filter	Enter the expression for filtering records that the scheduled job must reconcile. Sample value: `equalTo('__UID__','SEPT12USER1')` For information about the filters expressions that you can create and use, see ICF Filter Syntax in *Developing and Customizing Applications for Oracle Identity Governance*.
Incremental Recon Attribute	Name of the target system column that holds holds the timestamp at which the user record was modified. Default value: `lastModified`
Object Type	Type of object you want to reconcile. Default value: `User`
Latest Token	The parameter holds the value of the target system column that is specified as the value of the Incremental Recon Attribute parameter. The Latest Token parameter is used for internal purposes. By default, this value is empty. **Note:** Do not enter a value for this attribute. The reconciliation engine automatically enters a value in this attribute.
Scheduled Task Name	Name of the scheduled job. **Note:** For the scheduled job included with this connector, you must not change the value of this attribute. However, if you create a new job or create a copy of the job, then enter the unique name for that scheduled job as the value of this attribute.

Delete User Reconciliation Jobs

The following reconciliation jobs are available for reconciling data about deleted user accounts:

- DBUM Oracle Delete User Target Reconciliation: Use this reconciliation job to reconcile data about deleted user accounts from a Target application.
- DBUM Oracle Delete User Trusted Reconciliation: Use this reconciliation job to reconcile data about deleted user accounts from an Authoritative application.

The parameters for both these jobs are the same.

Table 3-12 Parameters of the Delete User Reconciliation Jobs for Oracle Database

Parameter	Description
Application Name	Name of the application you created for your target system. This value is the same as the value that you provided for the Application Name field while creating your target application. Do *not* modify this value.
Object Type	Type of object you want to reconcile. Default value: `User`

Reconciliation Jobs for Entitlements

The following jobs are available for reconciling entitlements:

- DBUM Oracle Privileges Lookup Reconciliation
- DBUM Oracle Profile Lookup Reconciliation
- DBUM Oracle Roles Lookup Reconciliation
- DBUM Oracle Tablespaces Lookup Reconciliation
- DBUM Oracle Temporary Tablespaces Lookup Reconciliation

These reconciliation jobs are available only for a Target application. The parameters for all the reconciliation jobs are the same.

Table 3-13 Parameters of the Reconciliation Jobs for Entitlements

Parameter	Description
Application Name	Name of the application you created for your target system. This value is the same as the value that you provided for the Application Name field while creating your target application. Do not modify this value.
Lookup Name	This parameter holds the name of the lookup definition that maps each lookup definition with the data source from which values must be fetched. Depending on the reconciliation job you are using, the default values are as follows: • For DBUM Oracle Privileges Lookup Reconciliation - `Lookup.DBUM.Oracle.Privileges` • For DBUM Oracle Profile Lookup Reconciliation - `Lookup.DBUM.Oracle.Profiles` • For DBUM Oracle Roles Lookup Reconciliation - `Lookup.DBUM.Oracle.Roles` • For DBUM Oracle Tablespaces Lookup Reconciliation - `Lookup.DBUM.Oracle.Tablespaces` • For DBUM Oracle Temporary Tablespaces Lookup Reconciliation - `Lookup.DBUM.Oracle.Temp.Tablespace`

Table 3-13 (Cont.) Parameters of the Reconciliation Jobs for Entitlements

Parameter	Description
Object Type	Enter the type of object whose values must be synchronized. Depending on the scheduled job you are using, the default values are as follows: • For DBUM Oracle Privileges Lookup Reconciliation - `__PRIVILEGES__` • For DBUM Oracle Profile Lookup Reconciliation - `__PROFILE__` • For DBUM Oracle Roles Lookup Reconciliation - `__ROLES__` • For DBUM Oracle Tablespaces Lookup Reconciliation - `__TABLESPACES__` • For DBUM Oracle Temporary Tablespaces Lookup Reconciliation - `__TEMPTABLESPACES__` **Note:** Do not change the value of this attribute.
Code Key Attribute	Enter the name of the connector or target system attribute that is used to populate the Code Key column of the lookup definition (specified as the value of the Lookup Name attribute). Default value: `__NAME__` **Note:** Do not change the value of this attribute.
Decode Attribute	Enter the name of the connector or target system attribute that is used to populate the Decode column of the lookup definition (specified as the value of the Lookup Name attribute). Default value: `__NAME__`

4

Configuring the Database User Management Connector for MySQL

While creating an application, you must configure connection-related parameters that the connector uses to connect Oracle Identity Governance with your target system and perform connector operations. In addition, you can view and edit attribute mappings between the process form fields in Oracle Identity Governance and target system columns, predefined correlation rules, situations and responses, and reconciliation jobs.

- Basic Configuration Parameters for MySQL
- Advanced Settings Parameters for MySQL
- Attribute Mappings for MySQL
- Correlation Rules for a MySQL Target Application
- Reconciliation Jobs for MySQL

4.1 Basic Configuration Parameters for MySQL

These are the connection-related parameters that Oracle Identity Governance requires to connect to MySQL.

Table 4-1 Basic Configuration Parameters for MySQL

Parameter	Mandatory?	Description
Connection Properties	No	Enter the connection properties for the target system database.
Connection URL	Yes	Enter the connection URL for your MySQL database. Default value: `jdbc:mysql://%h:%p/database`
Connector Server Name	No	If you created an IT resource of the type “Connector Server”, then enter its name.
Database Drivers	Yes	This parameter holds the name of the JDBC driver class. Default value: `com.mysql.jdbc.Driver`
Database Type	Yes	This parameter identifies the database type (such as Oracle or MySQL) and is used for loading respective scripts.

Table 4-1 (Cont.) Basic Configuration Parameters for MySQL

Parameter	Mandatory?	Description
Password	Yes	Enter the password for the user name of the target system account to be used for connector operations.
User	Yes	Enter the user name of the target system account to be used for connector operations. Sample value: `root`

4.2 Advanced Settings Parameters for MySQL

These are the configuration-related entries that the connector uses during reconciliation and provisioning operations.

> **Note:**
>
> Unless specified, the parameters in the table are applicable to both target and authoritative applications.

Table 4-2 Advanced Setting Parameters for MySQL

Parameter	Mandatory?	Description
Connector Name	Yes	This parameter holds the name of the connector class. **Value:** `org.identityconnectors.dbum.DBUMConnector`
Connector Package Name	Yes	This parameter holds the name of the connector bundle package. **Value:** `org.identityconnectors.dbum`
Connector Package Version	Yes	This parameter holds the version of the connector bundle class. **Value:** 1.0.1116
disableValuesSet	No	Enter the possible values for the disabled status of a user. **Default value:** `"EXPIRED & LOCKED","LOCKED","EXPIRED"`

Table 4-2 (Cont.) Advanced Setting Parameters for MySQL

Parameter	Mandatory?	Description
Reserve Keywords	No	Enter the list of words that are reserved and are not allowed to be used in the names of the connector artifacts **Default value:** `"DROP","INSERT","ALTER","CREATE","DELETE","UPDATE","GRANT","TRUNCATE","EXEC","TEMPORARY","TABLESPACE","DEFAULT","QUOTA","PROFILE","IDENTIFIED","EXTERNALLY","AS","GLOBALLY","REVOKE","ACCOUNT","UNLOCK","LOCK","CASCADE"` **Note:** This parameter is available only when you are creating a target application.
Unsupported Character Set	No	Enter the characters that are not allowed to be used in the names of the connector artifacts **Default value:** "&","--","~","`","\"" **Note:** This parameter is available only when you are creating a target application.
Pool Max Idle	No	Maximum number of idle objects in a pool. **Sample value:** `10`
Pool Max Size	No	Maximum number of connections that the pool can create. **Sample value:** `10`
Pool Max Wait	No	Maximum time, in milliseconds, the pool must wait for a free object to make itself available to be consumed for an operation. **Sample value:** `150000`
Pool Min Evict Idle Time	No	Minimum time, in milliseconds, the connector must wait before evicting an idle object. **Sample value:** `120000`
Pool Min Idle	No	Minimum number of idle objects in a pool. **Sample value:** `1`

4.3 Attribute Mappings for MySQL

The attribute mappings on the Schema page vary depending on whether you are creating a target application or an authoritative application.

- Attribute Mappings for a MySQL Target Application
- Attribute Mappings for a MySQL Authoritative Application

4.3.1 Attribute Mappings for a MySQL Target Application

The Schema page for a target application displays the default schema (provided by the connector) that maps Oracle Identity Governance attributes to target system columns. The connector uses these mappings during reconciliation and provisioning operations.

MySQL DB User Account Attributes

Table 4-3 lists the user-specific attribute mappings between the process form fields in Oracle Identity Governance and MySQL columns. The table also lists whether a specific attribute is used during provisioning or reconciliation and whether it is a matching key field for fetching records during reconciliation.

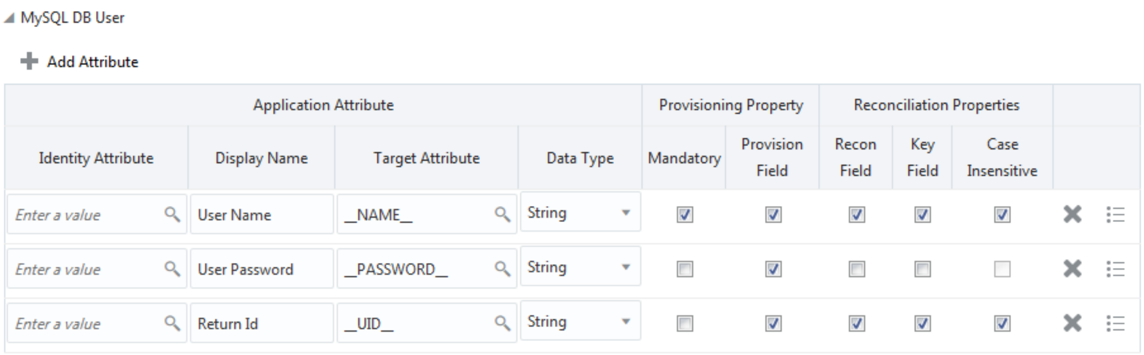
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application of *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 4-3 Default Attribute Mappings for MySQL DB User Account

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Provision Field?	Recon Field?	Key Field?	Case Insensitive?
Return Id	__UID__	String	No	Yes	Yes	Yes	Yes
Username	__NAME__	String	Yes	Yes	Yes	Yes	Yes
User Password	__PASSWORD__	String	No	Yes	No	No	No

Figure 4-1 shows the default User account attribute mappings.

Figure 4-1 Default Attribute Mappings for MySQL User Account



◢ MySQL DB User

➕ Add Attribute

Application Attribute				Provisioning Property		Reconciliation Properties			
Identity Attribute	Display Name	Target Attribute	Data Type	Mandatory	Provision Field	Recon Field	Key Field	Case Insensitive	
Enter a value	User Name	__NAME__	String	☑	☑	☑	☑	☑	✖ ☰
Enter a value	User Password	__PASSWORD__	String	☐	☑	☐	☐	☐	✖ ☰
Enter a value	Return Id	__UID__	String	☐	☑	☑	☑	☑	✖ ☰

Privilege List Entitlement Attributes

Table 4-4 lists the roles-specific attribute mappings between the process form fields in Oracle Identity Governance and MySQL columns. The table lists whether a given attribute is mandatory during provisioning. It also lists whether a given attribute is used during reconciliation and whether it is a matching key field for fetching records during reconciliation.

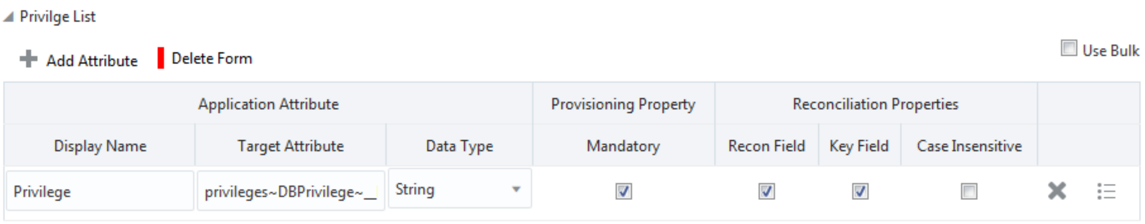
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application of *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 4-4 Default Attribute Mappings for MySQL Privilege List Entitlement

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Recon Field	Key Field?	Case Insensitive?
Privilege	privileges~DBPrivilege~__NAME__	String	Yes	Yes	Yes	No

Figure 4-2 shows the default Privilege List entitlement mapping.

Figure 4-2 Default Attribute Mappings for MySQL Privilege List Entitlement



◢ Privilge List

➕ Add Attribute ▌Delete Form ☐ Use Bulk

Application Attribute			Provisioning Property	Reconciliation Properties			
Display Name	Target Attribute	Data Type	Mandatory	Recon Field	Key Field	Case Insensitive	
Privilege	privileges~DBPrivilege~_	String	☑	☑	☑	☐	✖ ☰

4.3.2 Attribute Mappings for a MySQL Authoritative Application

The Schema page for an authoritative application displays the default schema (provided by the connector) that maps Oracle Identity Governance attributes to target system columns. The connector uses these mappings during reconciliation operations.

Table 4-5 lists the user-specific attribute mappings between the reconciliation fields in Oracle Identity Governance and Oracle Database columns. The table also lists the data type for a given attribute and specified whether it is a mandatory attribute for reconciliation.

If required, you can edit these attributes mappings by adding new attributes or deleting existing attributes on the Schema page as described in Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

You may use the default schema that has been set for you or update and change it before continuing to the next step.

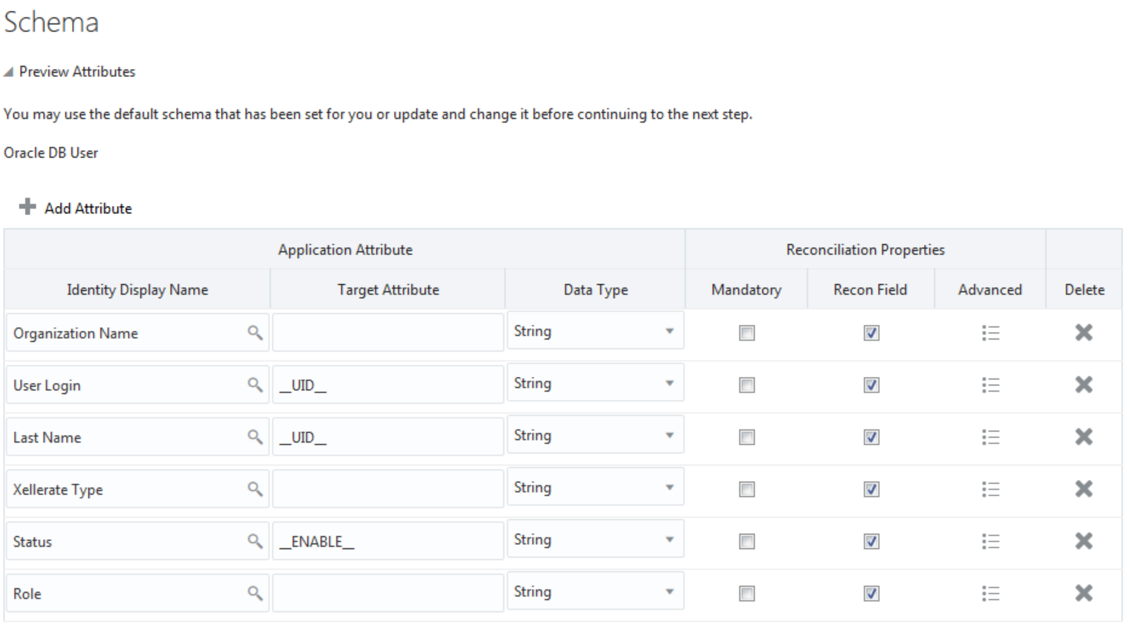
The Organization Name, Role, Xellerate Type, and Status identity attributes are mandatory fields on the OIG User form. They cannot be left blank during reconciliation. The target attribute mappings for these identity attributes are empty by default because there are no corresponding columns in the target system. Therefore, the connector provides default values (as listed in the “Default Value for Identity Display Name” column of Table 4-5) that it can use during reconciliation. For example, the default target attribute value for the Organization Name attribute is Xellerate Users. This implies that the connector reconciles all target system user accounts into the Xellerate Users organization in Oracle Identity Governance. Similarly, the default attribute value for Xellerate Type attribute is End-User, which implies that all reconciled user records are marked as end users.

Table 4-5 MySQL DB User Schema Attributes

Identity Display Name	Target Attribute	Data Type	Mandatory Reconciliation Property?	Recon Field?	Default Value for Identity Display Name
Organization Name	NA	String	No	Yes	Xellerate Users
Role	NA	String	No	Yes	Full-Time
User Login	__UID__	String	No	Yes	NA
Last Name	__NAME__	String	No	Yes	NA
Xellerate Type	NA	String	No	Yes	End-User
Status	NA	String	No	Yes	Active

Figure 3-4 shows the default User account attribute mappings.

Figure 4-3 Default Attribute Mappings for a MySQL User Account in an Authoritative Application



Schema

◢ Preview Attributes

You may use the default schema that has been set for you or update and change it before continuing to the next step.

Oracle DB User

✚ Add Attribute

Application Attribute			Reconciliation Properties			
Identity Display Name	Target Attribute	Data Type	Mandatory	Recon Field	Advanced	Delete
Organization Name		String	□	☑	☰	✖
User Login	__UID__	String	□	☑	☰	✖
Last Name	__UID__	String	□	☑	☰	✖
Xellerate Type		String	□	☑	☰	✖
Status	__ENABLE__	String	□	☑	☰	✖
Role		String	□	☑	☰	✖

4.4 Correlation Rules for MySQL

Learn about the predefined rules, responses and situations for Target and Authoritative applications. The connector use these rules and responses for performing reconciliation.

- Correlation Rules for a MySQL Target Application
- Correlation Rules for a MySQL Authoritative Application

4.4.1 Correlation Rules for a MySQL Target Application

The connector uses correlation rules to determine the identity to which Oracle Identity Governance must assign a resource.

Predefined Identity Correlation Rules

By default, the Database User Management connector provides a simple correlation rule when you create a Target application. The connector uses this correlation rule to compare the entries in Oracle Identity Governance repository and the target system repository, determine the difference between the two repositories, and apply the latest changes to Oracle Identity Governance.

Table 4-6 lists the default simple correlation rule for MySQL. If required, you can edit the default correlation rule or add new rules. You can create complex correlation rules also. For more information about adding or editing simple or complex correlation rules, see Creating a Target Application of *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 4-6 Predefined Identity Correlation Rule for MySQL

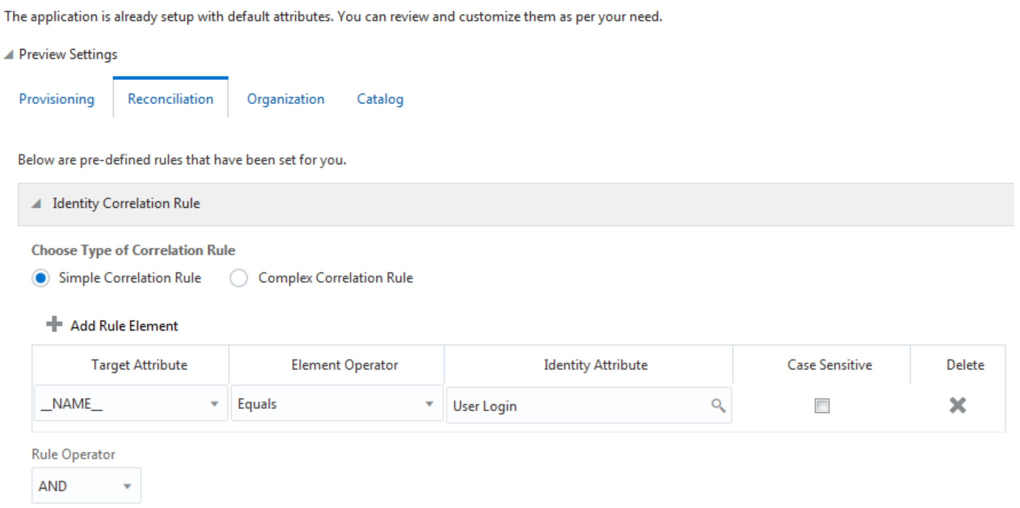
Target Attribute	Element Operator	Identity Attribute	Case Sensitive?
__NAME__	Equals	User Login	No

In this identity rule:

- __NAME__ is a single-valued attribute on the target system that identifies the user account.
- User Login is the field on the OIM User form.

Figure 3-5 shows the simple correlation rule for MySQL.

Figure 4-4 Simple Correlation Rule for MySQL



The application is already setup with default attributes. You can review and customize them as per your need.

Preview Settings

Below are pre-defined rules that have been set for you.

Identity Correlation Rule

Choose Type of Correlation Rule

● Simple Correlation Rule ○ Complex Correlation Rule

Target Attribute	Element Operator	Identity Attribute	Case Sensitive	Delete
NAME	Equals	User Login	□	✖

Rule Operator

AND

Predefined Situations and Responses

The Database User Management connector provides a default set of situations and responses when you create a Target application. These situations and responses specify the action that Oracle Identity Governance must take based on the result of a reconciliation event.

Table 4-7 lists the default situations and responses for MySQL. If required, you can edit these default situations and responses or add new ones. For more information about adding or editing situations and responses, see Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*

Table 4-7 Predefined Situations and Responses for MySQL

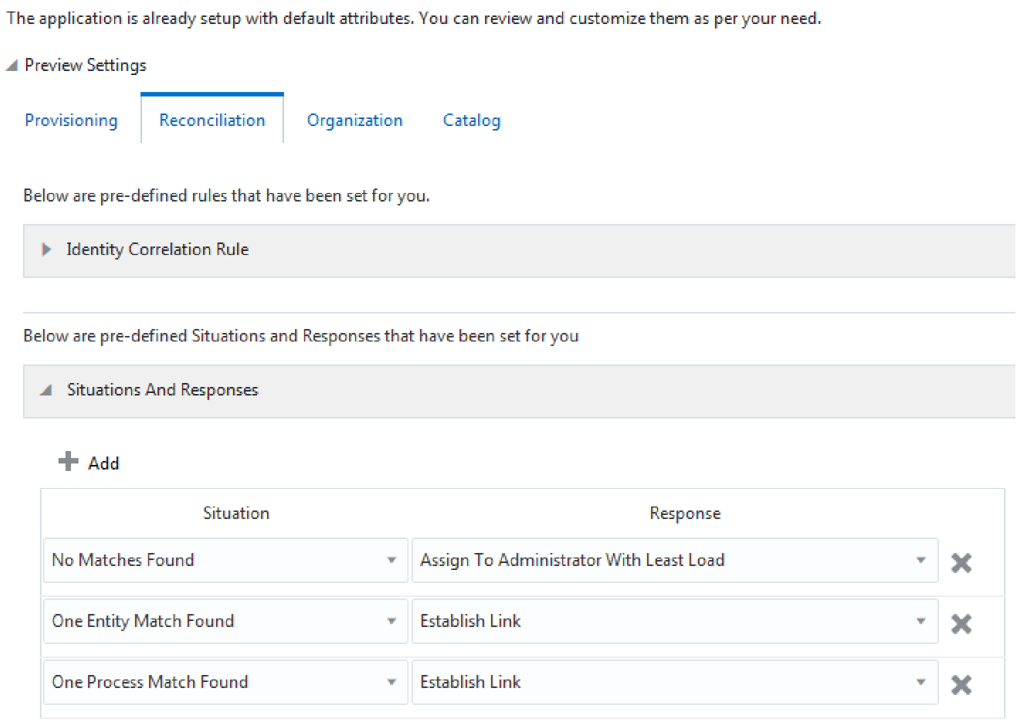
Situation	Response
No Matches Found	Assign to Administrator With Least Load
One Entity Match Found	Establish Link

Table 4-7 (Cont.) Predefined Situations and Responses for MySQL

Situation	Response
One Process Match Found	Establish Link

Figure 3-6 shows the situations and responses for MySQL that the connector provides by default.

Figure 4-5 Predefined Situations and Responses for MySQL



The application is already setup with default attributes. You can review and customize them as per your need.

Below are pre-defined rules that have been set for you.

Below are pre-defined Situations and Responses that have been set for you

Situation	Response
No Matches Found	Assign To Administrator With Least Load
One Entity Match Found	Establish Link
One Process Match Found	Establish Link

4.4.2 Correlation Rules for a MySQL Authoritative Application

When you create an Authoritative application, the connector uses correlation rules to determine the identity that must be reconciled into Oracle Identity Governance.

Predefined Identity Correlation Rules

By default, the Database User Management connector provides a simple correlation rule when you create an Authoritative application. The connector uses this correlation rule to compare the entries in Oracle Identity Governance repository and the target system repository, determine the difference between the two repositories, and apply the latest changes to Oracle Identity Governance.

Table 4-8 lists the default simple correlation rule for a MySQL authoritative application. If required, you can edit the default correlation rule or add new rules. You can create complex correlation rules also. For more information about adding or editing simple or complex correlation rules, see Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 4-8 Predefined Identity Correlation Rule for a MySQL Authoritative Application

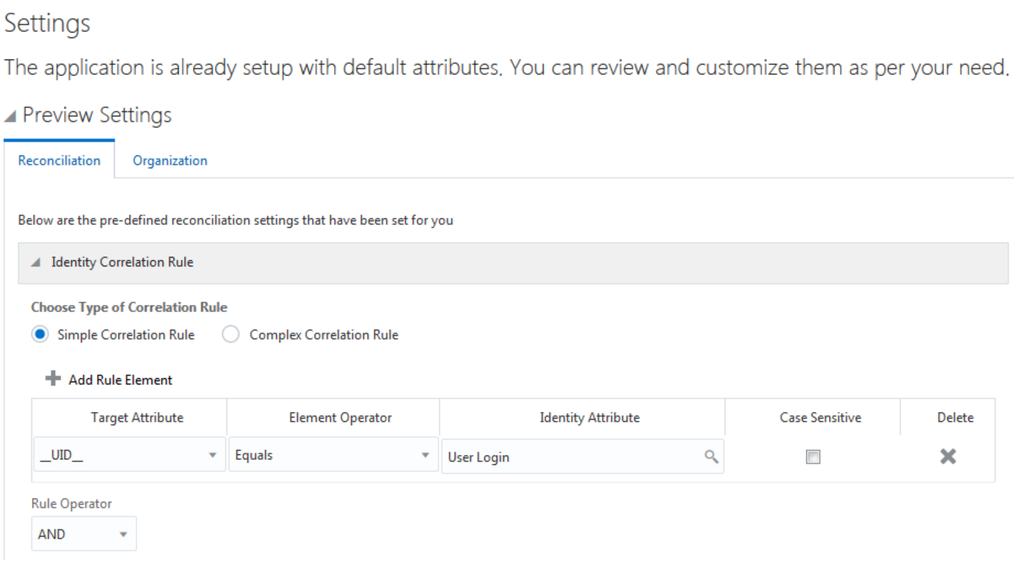
Target Attribute	Element Operator	Identity Attribute	Case Sensitive?
__UID__	Equals	User Login	No

In this identity rule:

- __UID__ is an attribute on the target system that uniquely identifies the user account.
- User Login is the field on the OIM User form.

Figure 3-7 shows the simple correlation rule for MySQL.

Figure 4-6 Simple Correlation Rule for a MySQL Authoritative Application



Settings

The application is already setup with default attributes. You can review and customize them as per your need.

◢ Preview Settings

Below are the pre-defined reconciliation settings that have been set for you

◢ Identity Correlation Rule

Choose Type of Correlation Rule

● Simple Correlation Rule ○ Complex Correlation Rule

Target Attribute	Element Operator	Identity Attribute	Case Sensitive	Delete
UID	Equals	User Login	□	✖

Rule Operator: AND

Predefined Situations and Responses

The Database User Management connector provides a default set of situations and responses when you create an Authoritative application. These situations and responses specify the action that Oracle Identity Governance must take based on the result of a reconciliation event.

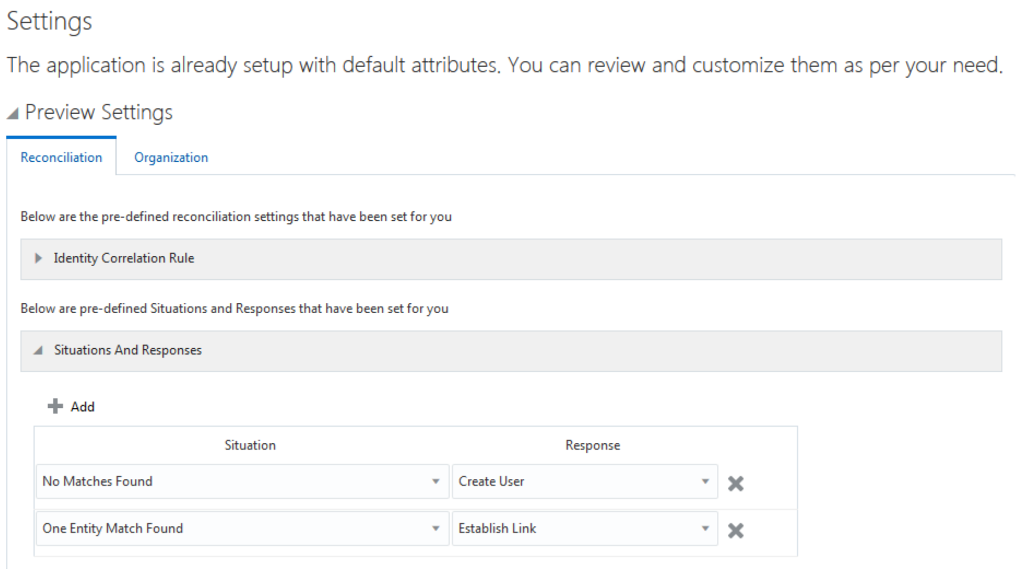
Table 4-9 lists the default situations and responses for MySQL. If required, you can edit these default situations and responses or add new ones. For more information about adding or editing situations and responses, see Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*

Table 4-9 Predefined Situations and Responses for a MySQL Authoritative Application

Situation	Response
No Matches Found	Create User
One Entity Match Found	Establish Link

Figure 3-8 shows the situations and responses for Oracle Database that the connector provides by default.

Figure 4-7 Predefined Situations and Responses for a MySQL Authoritative Application



Settings

The application is already setup with default attributes. You can review and customize them as per your need.

◢ Preview Settings

Below are the pre-defined reconciliation settings that have been set for you

▶ Identity Correlation Rule

Below are pre-defined Situations and Responses that have been set for you

◢ Situations And Responses

Situation	Response
No Matches Found	Create User
One Entity Match Found	Establish Link

4.5 Reconciliation Jobs for MySQL

These are the reconciliation jobs that are automatically created in Oracle Identity Governance after you create the application for your target system.

You can either use these predefined jobs or edit them to meet your requirements. Alternatively, you can create custom reconciliation jobs. For information about editing these predefined jobs or creating new ones, see Updating Reconciliation Jobs in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

User Reconciliation Jobs

The following reconciliation jobs are available for reconciling user data:

- DBUM MySQL User Target Reconciliation: Use this reconciliation job to reconcile user data from a Target application.
- DBUM MySQL User Trusted Reconciliation: Use this reconciliation job to reconcile user data from an Authoritative application.

The parameters for both these jobs are the same.

Table 4-10 Parameters of the User Reconciliation Jobs for MySQL

Parameter	Description
Application Name	Name of the application you created for your target system. This value is the same as the value that you provided for the Application Name field while creating your target application. Do not modify this value.
Batch Size	Enter the number of records that must be included in each batch fetched from the target system during reconciliation.
Filter	Enter the expression for filtering records that the scheduled job must reconcile. Sample value: `equalTo('__UID__','SEPT12USER1')` For information about the filters expressions that you can create and use, see ICF Filter Syntax in *Developing and Customizing Applications for Oracle Identity Governance*.
Object Type	Type of object you want to reconcile. Default value: `User`
Scheduled Task Name	Name of the scheduled job. **Note:** For the scheduled job included with this connector, you must not change the value of this attribute. However, if you create a new job or create a copy of the job, then enter the unique name for that scheduled job as the value of this attribute.

Delete User Reconciliation Jobs

The following reconciliation jobs are available for reconciling data about deleted user accounts:

- DBUM MySQL Delete User Target Reconciliation: Use this reconciliation job to reconcile data about deleted user accounts from a Target application.
- DBUM MySQL Delete User Trusted Reconciliation: Use this reconciliation job to reconcile data about deleted user accounts from an Authoritative application.

The parameters for both these jobs are the same.

Table 4-11 Parameters of the Delete User Reconciliation Jobs for MySQL

Parameter	Description
Application Name	Name of the application you created for your target system. This value is the same as the value that you provided for the Application Name field while creating your target application. Do not modify this value.
Object Type	Type of object you want to reconcile. Default value: `User`

Reconciliation Jobs for Entitlements

Use the DBUM MySQL Privilege Type Lookup Reconciliation job to reconcile the list of privileges from your target system. This job is available only for a Target application.

Table 4-12 Parameters of the DBUM MySQL Privilege TypeLookup Reconciliation Job

Parameter	Description
Application Name	Name of the application you created for your target system. This value is the same as the value that you provided for the Application Name field while creating your target application. Do not modify this value.
Lookup Name	This parameter holds the name of the lookup definition that maps each lookup definition with the data source from which values must be fetched. Default value: `Lookup.DBUM.MySQL.SchemaPrivileges`
Object Type	Enter the type of object whose values must be synchronized. Default value: `__PRIVILEGES__` **Note:** Do not change the value of this attribute.
Code Key Attribute	Enter the name of the connector or target system attribute that is used to populate the Code Key column of the lookup definition (specified as the value of the Lookup Name attribute). Default value: `__NAME__` **Note:** Do not change the value of this attribute.
Decode Attribute	Enter the name of the connector or target system attribute that is used to populate the Decode column of the lookup definition (specified as the value of the Lookup Name attribute). Default value: `__NAME__`

5

Performing the Postconfiguration Tasks

These are the tasks that you can perform after creating an application in Oracle Identity Governance.

- Configuring Oracle Identity Governance
- Harvesting Entitlements and Sync Catalog
- Managing Logging for Oracle Identity Governance
- Configuring the IT Resource for the Connector Server
- Creating the Administrator Account on Oracle Database Vault
- Localizing Field Labels in UI Forms for Database User Management Connector
- Configuring Secure Communication Between the Target System and Oracle Identity Governance

5.1 Configuring Oracle Identity Governance

During application creation, if you did not choose to create a default form, then you must create a UI form for the application that you created by using the connector.

> **Note:**
>
> Perform the procedures described in this section only if you did not choose to create the default form during creating the application.

The following topics describe the procedures to configure Oracle Identity Governance:

- Creating and Activating a Sandbox
- Creating a New UI Form
- Publishing a Sandbox
- Updating an Existing Application Instance with a New Form

5.1.1 Creating and Activating a Sandbox

You must create and activate a sandbox to begin using the customization and form management features. You can then publish the sandbox to make the customizations available to other users.

See Creating a Sandbox and Activating a Sandbox in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*.

5.1.2 Creating a New UI Form

You can use Form Designer in Oracle Identity System Administration to create and manage application instance forms.

See Creating Forms By Using the Form Designer in *Oracle Fusion Middleware Administering Oracle Identity Governance*.

While creating the UI form, ensure that you select the resource object corresponding to the newly created application that you want to associate the form with. In addition, select the **Generate Entitlement Forms** check box.

5.1.3 Publishing a Sandbox

Before publishing a sandbox, perform this procedure as a best practice to validate all sandbox changes made till this stage as it is difficult to revert the changes after a sandbox is published.

1. In Identity System Administration, deactivate the sandbox.
2. Log out of Identity System Administration.
3. Log in to Identity Self Service using the xelsysadm user credentials and then activate the sandbox that you deactivated in Step 1.
4. In the Catalog, ensure that the application instance form for your resource appears with correct fields.
5. Publish the sandbox. See Publishing a Sandbox in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*.

5.1.4 Updating an Existing Application Instance with a New Form

For any changes that you do in the schema of your application in Identity Self Service, you must create a new UI form and update the changes in an application instance.

To update an existing application instance with a new form:

1. Create and activate a sandbox.
2. Create a new UI form for the resource.
3. Open the existing application instance.
4. In the Form field, select the new UI form that you created.
5. Save the application instance.
6. Publish the sandbox.

See Also:

- Creating a Sandbox and Activating a Sandbox in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*
- Creating Forms By Using the Form Designer in *Oracle Fusion Middleware Administering Oracle Identity Governance*
- Publishing a Sandbox in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*

5.2 Harvesting Entitlements and Sync Catalog

You can populate Entitlement schema from child process form table, and harvest roles, application instances, and entitlements into catalog. You can also load catalog metadata.

To harvest entitlements and sync catalog:

1. Depending on the target system that you are using, run the scheduled jobs for lookup field synchronization as follows:

   For Oracle Database: Run the reconciliation jobs for entitlements listed in Reconciliation Jobs for Oracle Database.

   For MySQL: Run the reconciliation jobs for entitlements listed in Reconciliation Jobs for MySQL.

2. Run the Entitlement List scheduled job to populate Entitlement Assignment schema from child process form table.
3. Run the Catalog Synchronization Job scheduled job.

See Also:

Predefined Scheduled Tasks in *Oracle Fusion Middleware Administering Oracle Identity Governance* for a description of the Entitlement List and Catalog Synchronization Job scheduled jobs

5.3 Managing Logging for Oracle Identity Governance

Oracle Identity Governance uses the Oracle Diagnostic Logging (ODL) logging service for recording all types of events pertaining to the connector.

The following topics provide detailed information about logging:

- Understanding Log Levels
- Enabling Logging

5.3.1 Understanding Log Levels

When you enable logging, Oracle Identity Governance automatically stores in a log file information about events that occur during the course of provisioning and reconciliation operations.

ODL is the principle logging service used by Oracle Identity Governance and is based on java.util.logger. To specify the type of event for which you want logging to take place, you can set the log level to one of the following:

- SEVERE.intValue()+100

  This level enables logging of information about fatal errors.
- SEVERE

  This level enables logging of information about errors that might allow Oracle Identity Governance to continue running.
- WARNING

  This level enables logging of information about potentially harmful situations.
- INFO

  This level enables logging of messages that highlight the progress of the application.
- CONFIG

  This level enables logging of information about fine-grained events that are useful for debugging.
- FINE, FINER, FINEST

  These levels enable logging of information about fine-grained events, where FINEST logs information about all events.

These message types are mapped to ODL message type and level combinations as shown in Table 5-2.

Table 5-1 Log Levels and ODL Message Type:Level Combinations

Java Level	ODL Message Type:Level
SEVERE.intValue()+100	INCIDENT_ERROR:1
SEVERE	ERROR:1
WARNING	WARNING:1
INFO	NOTIFICATION:1
CONFIG	NOTIFICATION:16
FINE	TRACE:1
FINER	TRACE:16

Table 5-2 Log Levels and ODL Message Type:Level Combinations

Java Level	ODL Message Type:Level
SEVERE.intValue()+100	INCIDENT_ERROR:1

Table 5-2 (Cont.) Log Levels and ODL Message Type:Level Combinations

Java Level	ODL Message Type:Level
SEVERE	ERROR:1
WARNING	WARNING:1
INFO	NOTIFICATION:1
CONFIG	NOTIFICATION:16
FINE	TRACE:1
FINER	TRACE:16
FINEST	TRACE:32

The configuration file for OJDL is logging.xml, which is located at the following path:

DOMAIN_HOME/config/fmwconfig/servers/*OIM_SERVER*/logging.xml

Here, *DOMAIN_HOME* and *OIM_SERVER* are the domain name and server name specified during the installation of Oracle Identity Governance.

5.3.2 Enabling Logging

You can enable logging in Oracle WebLogic Server by updating the logging.xml file.

To enable logging:

1. Edit the logging.xml file as follows:

   a. Add the following blocks in the file:

   ```
   <log_handler name='db-um-handler' level='[LOG_LEVEL]'
   class='oracle.core.ojdl.logging.ODLHandlerFactory'>
   <property name='logreader:' value='off'/>
        <property name='path' value='[FILE_NAME]'/>
        <property name='format' value='ODL-Text'/>
        <property name='useThreadName' value='true'/>
        <property name='locale' value='en'/>
        <property name='maxFileSize' value='5242880'/>
        <property name='maxLogSize' value='52428800'/>
        <property name='encoding' value='UTF-8'/>
      </log_handler>

   <logger name="ORG.IDENTITYCONNECTORS.DBUM" level="[LOG_LEVEL]"
   useParentHandlers="false">
        <handler name="db-um-handler"/>
        <handler name="console-handler"/>
      </logger>
   ```

   b. Replace all occurrences of **`[LOG_LEVEL]`** with the ODL message type and level combination that you require. Table 5-2 lists the supported message type and level combinations.

   Similarly, replace **`[FILE_NAME]`** with the full path and name of the log file in which you want log messages to be recorded.

   The following blocks show sample values for **`[LOG_LEVEL]`** and **`[FILE_NAME]`** :

```
<log_handler name='db-um-handler' level='NOTIFICATION:1'
class='oracle.core.ojdl.logging.ODLHandlerFactory'>
<property name='logreader:' value='off'/>
     <property name='path'
value='F:\MyMachine\middleware\user_projects\domains\base_domain1\servers
\oim_server1\logs\oim_server1-diagnostic-1.log'/>
     <property name='format' value='ODL-Text'/>
     <property name='useThreadName' value='true'/>
     <property name='locale' value='en'/>
     <property name='maxFileSize' value='5242880'/>
     <property name='maxLogSize' value='52428800'/>
     <property name='encoding' value='UTF-8'/>
   </log_handler>

<logger name="oracle.iam.connectors.icfcommon" level="NOTIFICATION:1"
useParentHandlers="false">
     <handler name="db-um-handler"/>
   </logger>
<logger name="ORG.IDENTITYCONNECTORS.DBUM" level="NOTIFICATION:1"
useParentHandlers="false">
     <handler name="db-um-handler"/>
   </logger>
```

With these sample values, when you use Oracle Identity Governance, all messages generated for this connector that are of a log level equal to or higher than the `NOTIFICATION:1` level are recorded in the specified file.

2. Save and close the file.
3. Restart the application server.

5.4 Configuring the IT Resource for the Connector Server

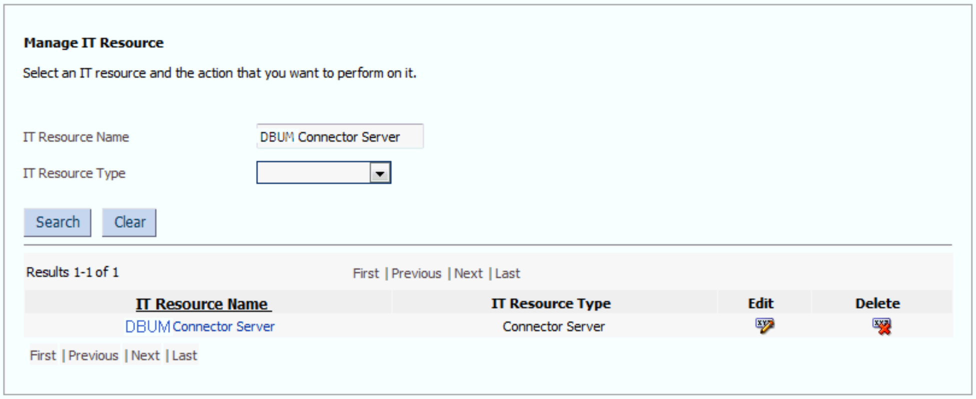
If you have used the Connector Server, you must configure values for the parameters of the Connector Server IT resource.

> **Note:**
>
> This procedure is optional and is required only when the Connector Server is being used.

To configure or modify the IT resource for the Connector Server:

1. Log in to Oracle Identity System Administration.
2. Create and activate a sandbox.
3. In the left pane, under Configuration, click **IT Resource**.
4. In the IT Resource Name field on the Manage IT Resource page, enter `DBUM Connector Server` and then click **Search.** Figure 5-1 shows the Manage IT Resource page.

Figure 5-1   Manage IT Resource Page for Connector Server IT Resource



Manage IT Resource

Select an IT resource and the action that you want to perform on it.

IT Resource Name	DBUM Connector Server
IT Resource Type	

Results 1-1 of 1

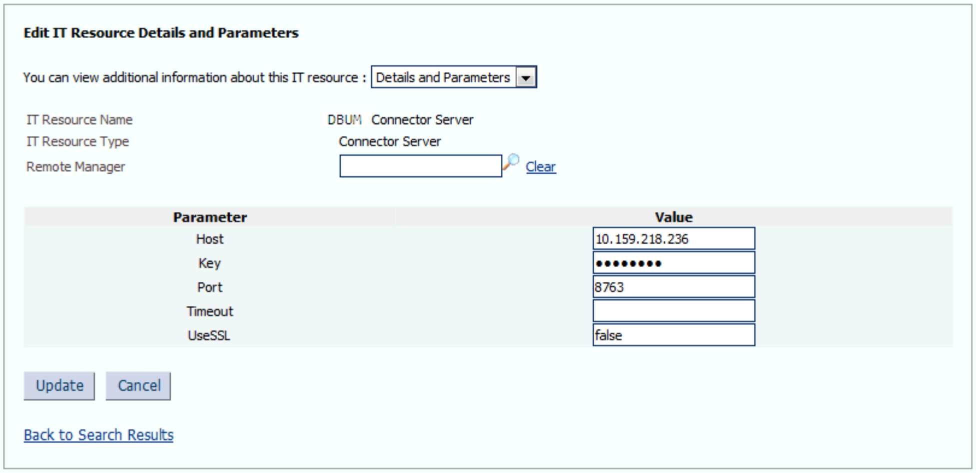
First | Previous | Next | Last

IT Resource Name	IT Resource Type	Edit	Delete
DBUM Connector Server	Connector Server		

First | Previous | Next | Last

5. Click the edit icon corresponding to the Connector Server IT resource.
6. From the list at the top of the page, select **Details and Parameters**.
7. Specify values for the parameters of the Connector Server IT resource. Figure 5-2 shows the Edit IT Resource Details and Parameters page.

Figure 5-2   Edit IT Resource Details and Parameters Page for the Connector Server IT Resource



Edit IT Resource Details and Parameters

You can view additional information about this IT resource : Details and Parameters

IT Resource Name	DBUM Connector Server
IT Resource Type	Connector Server
Remote Manager	

Parameter	Value
Host	10.159.218.236
Key	••••••••
Port	8763
Timeout	
UseSSL	false

Table 5-3 provides information about the parameters of the IT resource.

Table 5-3   Parameters of the IT Resource for the Database User Management Connector Server

Parameter	Description
Host	Enter the host name or IP address of the computer hosting the Connector Server. Sample value: `HostName`
Key	Enter the key for the Connector Server.
Port	Enter the number of the port at which the Connector Server is listening. Default value: `8763`

Table 5-3 (Cont.) Parameters of the IT Resource for the Database User Management Connector Server

Parameter	Description
Timeout	Enter an integer value which specifies the number of milliseconds after which the connection between the Connector Server and Oracle Identity Governance times out. If the value is zero or if no value is specified, the timeout is unlimited. Sample value: `0` (recommended value)
UseSSL	Enter `true` to specify that you will configure SSL between Oracle Identity Governance and the Connector Server. Otherwise, enter `false.` Default value: `false` **See Also:** Configuring Secure Communication Between the Target System and Oracle Identity Governance for information about enabling SSL.

8. To save the values, click **Update**.

5.5 Creating the Administrator Account on Oracle Database Vault

You must create an administrator account on Oracle Database Vault. This account is used by the connector for performing reconciliation and provisioning operations on Oracle Database Vault realms.

> **Note:**
>
> Perform the procedure described in this section only if you have Oracle Database Vault installed and you want to configure the connector for provisioning and reconciling authorization to Oracle Database Vault realms.

To create the administrator account on Oracle Database Vault:

1. Log in to Oracle Database Vault as a user with the DV_ACCTMGR privilege.
2. Create the administrator account by running the following command:

   ```
   CREATE USER USERNAME IDENTIFIED BY PASSWORD;
   ```

3. Log out and then log in as a user with the DV_OWNER privilege.
4. Grant access to Oracle Database Vault and Data Dictionary realms by running the following commands:

   ```
   exec DVSYS.DBMS_MACADM.ADD_AUTH_TO_REALM('Database Vault Account
   Management','USERNAME','Enabled',1)
   exec DVSYS.DBMS_MACADM.ADD_AUTH_TO_REALM('Oracle Data
   Dictionary','USERNAME','Enabled',1)
   ```

5. Grant the DV_ADMIN and DV_SECANALYST privileges.
6. Log in as a user with the DV_ACCTMGR privilege.
7. Grant the DV_SECANALYST privilege.

8. Log in as SYS and grant the following privileges (run the command):

   ```
   GRANT ANY OBJECT PRIVILEGE
   GRANT ANY PRIVILEGE
   GRANT ANY ROLE
   UNLIMITED TABLESPACE
   with ADMIN OPTION
   to USERNAME
   ```

5.6 Localizing Field Labels in UI Forms for Database User Management Connector

You can localize UI form field labels by using the resource bundle corresponding to the language you want to use. The resource bundles are available in the connector installation media.

> **Note:**
>
> Perform the procedure described in this section only if you are using Oracle Identity Manager release 11.1.2.*x* or later and you want to localize UI form field labels.

To localize field label that you add to in UI forms:

1. Log in to Oracle Enterprise Manager.
2. In the left pane, expand **Application Deployments** and then select **oracle.iam.console.identity.sysadmin.ear**.
3. In the right pane, from the Application Deployment list, select **MDS Configuration**.
4. On the MDS Configuration page, click **Export** and save the archive to the local computer.
5. Extract the contents of the archive, and open the following file in a text editor:

   SAVED_LOCATION\xliffBundles\oracle\iam\ui\runtime\BizEditorBundle.xlf
6. Edit the BizEditorBundle.xlf file in the following manner:
   a. Search for the following text:

      ```
      <file source-language="en"
      original="/xliffBundles/oracle/iam/ui/runtime/BizEditorBundle.xlf"
      datatype="x-oracle-adf">
      ```
   b. Replace with the following text:

      ```
      <file source-language="en" target-language="LANG_CODE"
      original="/xliffBundles/oracle/iam/ui/runtime/BizEditorBundle.xlf"
      datatype="x-oracle-adf">
      ```

      In this text, replace *LANG_CODE* with the code of the language that you want to localize the form field labels. The following is a sample value for localizing the form field labels in Japanese:

```
<file source-language="en" target-language="ja"
original="/xliffBundles/oracle/iam/ui/runtime/BizEditorBundle.xlf"
datatype="x-oracle-adf">
```

c. Search for the application instance code. This procedure shows a sample edit for Oracle Database application instance. The original code is:

```
<trans-unit id="$
{adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResourceBundl
e']
['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity.userEO.
UD_DB_ORA_U_USERNAME__c_description']}">
<source>Username</source>
</target>
</trans-unit>
<trans-unit
id="sessiondef.oracle.iam.ui.runtime.form.model.OracleDBForm.entity.Oracl
eDBForm.UD_DB_ORA_U_USERNAME__c_LABEL">
<source>Username</source>
</target>
</trans-unit>
```

d. Open the resource file from the connector package, for example DB-UM_ja.properties, and get the value of the attribute from the file, for example, global.udf.UD_DB_ORA_U_USERNAME=\u30E6\u30FC\u30B6\u30FC\u540D.

e. Replace the original code shown in Step 6.b with the following:

```
<trans-unit id="$
{adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResourceBundl
e']
['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity.userEO.
UD_DB_ORA_U_USERNAME__c_description']}">
<source>Username</source>
<target>\u30E6\u30FC\u30B6\u30FC\u540D</target>
</trans-unit>
<trans-unit
id="sessiondef.oracle.iam.ui.runtime.form.model.OracleDBForm.entity.Oracl
eDBForm.UD_DB_ORA_U_USERNAME__c_LABEL">
<source>Username</source>
<target>\u30E6\u30FC\u30B6\u30FC\u540D</target>
</trans-unit>
```

f. Repeat Steps 6.a through 6.d for all attributes of the process form.

g. Save the file as BizEditorBundle_*LANG_CODE*.xlf. In this file name, replace *LANG_CODE* with the code of the language to which you are localizing.

   Sample file name: BizEditorBundle_ja.xlf.

7. Repackage the ZIP file and import it into MDS.

> **See Also:**
>
> Deploying and Undeploying Customizations in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance* for more information about exporting and importing metadata files

8. Log out of and log in to Oracle Identity Governance.

5.7 Configuring Secure Communication Between the Target System and Oracle Identity Governance

You must configure SSL to secure the communication between your target system and Oracle Identity Governance.

- Configuring Secure Communication Between Oracle Database and Oracle Identity Governance
- Configuring Secure Communication Between MySQL and Oracle Identity Governance

5.7.1 Configuring Secure Communication Between Oracle Database and Oracle Identity Governance

It is recommended that you perform the procedure described in this section to configure secure communication between Oracle Database and Oracle Identity Governance.

To secure communication between Oracle Database and Oracle Identity Governance, you can perform either one or both of the following procedures:

- Configuring Data Encryption and Integrity in Oracle Database
- Configuring SSL Communication in Oracle Database

5.7.1.1 Configuring Data Encryption and Integrity in Oracle Database

You can protect data against active attacks and ensure data privacy by configuring native Oracle Net Services data encryption and integrity for Oracle Advanced Security.

To configure data encryption and integrity, see Data Encryption in *Oracle Database Advanced Security Administrator's Guide*.

5.7.1.2 Configuring SSL Communication in Oracle Database

You configure SSL to secure data communication between Oracle Identity Governance and the target system.

To enable SSL communication:

1. See *Oracle Database Advanced Security Administrator's Guide* for information about enabling SSL communication between Oracle Database and Oracle Identity Governance.

   Export the certificate on the Oracle Database host computer.
2. Copy the certificate to Oracle Identity Governance.
3. Import the certificate into the JVM truststore of the application server on which Oracle Identity Governance is running.

   To import the certificate into the truststore, run the following command:

   ```
   ..\..\bin\keytool -import -file FILE_LOCATION -keystore TRUSTSTORE_LOCATION
   -storepass TRUSTSTORE_PASSWORD -trustcacerts -alias ALIAS
   ```

In this command:

- Replace *FILE_LOCATION* with the full path and name of the certificate file.
- Replace *ALIAS* with an alias for the certificate.
- Replace *TRUSTSTORE_PASSWORD* with a password for the truststore.
- Replace *TRUSTSTORE_LOCATION* with one of the truststore paths from Table 5-4. This table shows the location of the truststore for each of the supported application servers.

> **Note:**
>
> In an Oracle Identity Governance cluster, import the file into the truststore on each node of the cluster.

Table 5-4 Truststore Locations on Supported Application Servers

Application Server	Truststore Location
Oracle WebLogic Server	• If you are using Oracle jrockit_R27.3.1-jdk, then import the certificate into the keystore in the following directory: *JROCKIT_HOME*/jre/lib/security • If you are using the default Oracle WebLogic Server JDK, then import the certificate into the keystore in following directory: *WEBLOGIC_HOME*/java/jre/lib/security/cacerts • If you are using a JDK other than Oracle jrockit_R27.3.1-jdk or Oracle WebLogic Server JDK, then import the certificate into your keystore at the following directory: *JAVA_HOME*/jre/lib/security/cacerts

4. To enable secure communication between Oracle Database and Oracle Identity Governance, set the value of the UseSSL parameter of the IT resource for Connector Server to `true`. To configure SSL for the Connector Server, see Configuring SSL for Java Connector Server in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*.

5.7.2 Configuring Secure Communication Between MySQL and Oracle Identity Governance

It is recommended that you perform the procedure described in this section to configure secure communication between your target system and Oracle Identity Governance.

To secure communication between MySQL and Oracle Identity Governance:

1. See MySQL documentation for information about enabling SSL communication between MySQL and a client system. In this context, the client is Oracle Identity Governance.
2. Export the certificate on the MySQL host computer.

3. Restart the MySQL database service by using the certificate exported in the preceding step. See MySQL documentation for information on restarting the database service.

4. Copy the ca-cert.pem and client-cert.pem certificates to the Oracle Identity Governance host computer.

5. Import the certificates into the JVM truststore of the application server on which Oracle Identity Governance is running.

   To import the certificates into the truststore, run the following command for each certificate:

   ```
   keytool -import -file FILE_LOCATION -keystore TRUSTSTORE_LOCATION -storepass
   TRUSTSTORE_PASSWORD -trustcacerts -alias ALIAS
   ```

   In this command:

   - Replace *FILE_LOCATION* with the full path and name of the certificate file.
   - Replace *ALIAS* with an alias for the certificate.
   - Replace *TRUSTSTORE_PASSWORD* with a password for the truststore.
   - Replace *TRUSTSTORE_LOCATION* with one of the truststore paths from Table 5-5. This table shows the location of the truststore for each of the supported application servers.

   > **Note:**
   >
   > In an Oracle Identity Governance cluster, import the file into the truststore on each node of the cluster.

Table 5-5 Truststore Locations on Supported Application Servers

Application Server	Truststore Location
Oracle WebLogic Server	• If you are using Oracle jrockit_R27.3.1-jdk, then import the certificate into the keystore in the following directory: *JROCKIT_HOME*/jre/lib/security • If you are using the default Oracle WebLogic Server JDK, then import the certificate into the keystore in following directory: *WEBLOGIC_HOME*/java/jre/lib/security/cacerts • If you are using a JDK other than Oracle jrockit_R27.3.1-jdk or Oracle WebLogic Server JDK, then import the certificate into your keystore at the following directory: *JAVA_HOME*/jre/lib/security/cacerts

6. To enable secure communication between MySQL and Oracle Identity Governance, set the value of the UseSSL parameter of the IT resource for Connector Server to `true`. To configure SSL for the Connector Server, see Configuring SSL for Java Connector Server in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*.

6

Using the Database User Management Connector

You can use the Database User Management connector for performing reconciliation and provisioning operations after configuring your application to meet your requirements.

- Guidelines on Configuring Reconciliation
- Configuring Reconciliation
- Configuring Reconciliation Jobs
- Guidelines on Performing Provisioning Operations
- Performing Provisioning Operations
- Uninstalling the Connector

6.1 Guidelines on Configuring Reconciliation

These are the guidelines that you must apply while configuring reconciliation for Oracle Database and MySQL.

- Before you perform a target resource reconciliation run, you must synchronize the lookup definitions with the lookup fields of the target system. In other words, the scheduled job for lookup field synchronization must be run before user reconciliation runs.
- After you configure batched reconciliation, if reconciliation fails during a batched reconciliation run, then rerun the scheduled job without changing the values of the task attributes.

6.2 Configuring Reconciliation

You can configure the connector to specify the type of reconciliation and its schedule.

- Configuring Reconciliation for Oracle Database
- Configuring Reconciliation for MySQL

6.2.1 Configuring Reconciliation for Oracle Database

You can configure the connector to specify the type of reconciliation and its schedule.

- Performing Full and Incremental Reconciliation from Oracle Database
- Performing Limited Reconciliation from Oracle Database
- Performing Batched Reconciliation from Oracle Database

6.2.1.1 Performing Full and Incremental Reconciliation from Oracle Database

Full reconciliation involves reconciling all existing user records from the target system into Oracle Identity Governance. After you create the application, you must first perform full reconciliation.

At the end of the reconciliation run, the Latest Token parameter of the reconciliation job for user record reconciliation is automatically updated. From the next reconciliation run onward, only records created after this time stamp are considered for reconciliation. This is incremental reconciliation.

You can switch from incremental reconciliation to full reconciliation whenever you want to ensure that all target system records are reconciled in Oracle Identity Governance.

To perform a full reconciliation run, remove (delete) any value currently assigned to the Latest Token and Filter parameters and run one of the following reconciliation jobs:

- For an Oracle Database Target application: DBUM Oracle User Target Reconciliation
- For an Oracle Database Authoritative Application: DBUM Oracle User Trusted Reconciliation

See Reconciliation Jobs for Oracle Database for information about these reconciliation jobs.

For example, the Incremental Recon Attribute maps to the CREATED column in the DBA_USERS table. After the first full reconciliation run, the Latest Token parameter gets populated accordingly. In subsequent reconciliation runs, the connector fetches only the user records that are created after the timestamp in the Latest Token parameter. Users updated after the time-stamp are not fetched.

6.2.1.2 Performing Limited Reconciliation from Oracle Database

You can perform limited reconciliation by creating filters for the reconciliation module, and reconcile records from the target system based on a specified filter criterion.

By default, all target system records that are added or modified after the last reconciliation run are reconciled during the current reconciliation run. You can customize this process by specifying the subset of added or modified target system records that must be reconciled.

This connector provides a Filter attribute (a scheduled task attribute) that allows you to use any of the DBUM resource attributes to filter the target system records. You can apply filters to the parent parameters in the reconciliation query file stored in a JAR file in the bundle directory of the connector installation media. For example, to locate the reconciliation query file, you can extract the `bundle/org.identityconnectors.dbum-12.3.0.jar` file and open `scripts/oracle/Search.queries`.

The following table provides a list of parent parameters that can be used with the Filter attribute of the scheduled jobs:

Parameter	Description
__UID__	Unique identity representing the user This parameter is mapped to USERNAME or __NAME__ connector attribute.
authType	Authentication type of the user account The value of this parameter must be `PASSWORD`.
tablespace	Default tablespace for user operations
defaultQuota	Quota for user operations on default tablespace If no value is specified, the quota is set to unlimited.
globalDN	Unique name that identifies a user across an enterprise, if the authentication type is GLOBAL
__ENABLE__	Status of the user account The user is disabled if the value is one of following: LOCKED, EXPIRED, or LOCKED & EXPIRED The list of values for the disabled status is provided in the Lookup.DBUM.Oracle.Configuration lookup definition.
tempTableSpace	Temporary tablespace for user operations Quota is always unlimited on temporary tablespace.
profile	Profile of the user account
lastModified	Last modified time-stamp This parameter is used for incremental reconciliation operations.

For detailed information about ICF Filters, see ICF Filter Syntax in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*.

6.2.1.3 Performing Batched Reconciliation from Oracle Database

You can perform batched reconciliation to reconcile a specific number of records from the target system into Oracle Identity Governance.

During a reconciliation run, all changes in the target system records are reconciled into Oracle Identity Governance. Depending on the number of records to be reconciled, this process may require a large amount of time. In addition, if the connection breaks during reconciliation, then the process would take longer to complete. You can configure batched reconciliation to avoid these problems.

To configure batched reconciliation, you must specify value for the Batch Size reconciliation job parameter. Use this parameter to specify the number of records that must be included in each batch. By default, this value is empty.

If you specify a value other than `All`, then some of the newly added or modified user records may not get reconciled during the current reconciliation run. The following example illustrates this:

Suppose you specify the Batch Size value as `200` while configuring the scheduled jobs. Suppose that 314 user records were created or modified after the last reconciliation run. Of these 314 records, only 200 records would be reconciled during the current reconciliation run. The remaining 114 records would be reconciled during the next reconciliation run.

You specify values for the Batch Size parameter by following the instructions described in Configuring Reconciliation Jobs.

6.2.2 Configuring Reconciliation for MySQL

You can configure the connector to specify the type of reconciliation and its schedule.

- Performing Full Reconciliation from MySQL
- Performing Limited Reconciliation from MySQL
- Performing Batched Reconciliation from MySQL

6.2.2.1 Performing Full Reconciliation from MySQL

Full reconciliation involves reconciling all existing user records from the target system into Oracle Identity Governance. After you create the application, you must first perform full reconciliation.

To perform a full reconciliation run, remove (delete) any value currently assigned to the Filter parameter and run one of the following reconciliation jobs:

- For a MySQL Target application: DBUM MySQL User Target Reconciliation
- For a MySQL Authoritative application: DBUM MySQL User Trusted Reconciliation

See Reconciliation Jobs for MySQL for more information about these scheduled jobs.

6.2.2.2 Performing Limited Reconciliation from MySQL

You can perform limited reconciliation by creating filters for the reconciliation module, and reconcile records from the target system based on a specified filter criterion.

By default, all target system records that are added or modified after the last reconciliation run are reconciled during the current reconciliation run. You can customize this process by specifying the subset of added or modified target system records that must be reconciled.

This connector provides a Filter attribute (a scheduled task attribute) that allows you to use any of the DBUM resource attributes to filter the target system records. You can apply filters to the parent parameters in the reconciliation query file stored in a JAR file in the bundle directory of the connector installation media. For example, to locate the reconciliation query file, you can extract the `bundle/org.identityconnectors.dbum-12.3.0.jar` file and open `scripts/mysql/Search.queries`.

The following table provides the description of the parent parameter that can be used with the Filter attribute of the scheduled jobs:

Parameter	Description
__UID__	Unique identity representing the user This parameter is mapped to USERNAME or __NAME__ connector attribute.

For detailed information about ICF Filters, see ICF Filter Syntax in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*.

6.2.2.3 Performing Batched Reconciliation from MySQL

You can perform batched reconciliation to reconcile a specific number of records from the target system into Oracle Identity Governance.

During a reconciliation run, all changes in the target system records are reconciled into Oracle Identity Governance. Depending on the number of records to be reconciled, this process may require a large amount of time. In addition, if the connection breaks during reconciliation, then the process would take longer to complete. You can configure batched reconciliation to avoid these problems.

To configure batched reconciliation, you must specify value for the Batch Size reconciliation scheduled job attribute. Use this attribute to specify the number of records that must be included in each batch. By default, this value is empty.

If you specify a value other than `All`, then some of the newly added or modified user records may not get reconciled during the current reconciliation run. The following example illustrates this:

Suppose you specify the Batch Size value as `200` while configuring the scheduled jobs. Suppose that 314 user records were created or modified after the last reconciliation run. Of these 314 records, only 200 records would be reconciled during the current reconciliation run. The remaining 114 records would be reconciled during the next reconciliation run.

You specify values for the Batch Size attribute by following the instructions described in Configuring Reconciliation Jobs.

6.3 Configuring Reconciliation Jobs

Configure reconciliation jobs to perform reconciliation runs that check for new information on your target system periodically and replicates the data in Oracle Identity Governance.

You can apply this procedure to configure the reconciliation jobs for users and entitlements.

To configure a reconciliation job:

1. Log in to Identity System Administration.
2. In the left pane, under System Management, click **Scheduler**.
3. Search for and open the scheduled job as follows:
   a. In the Search field, enter the name of the scheduled job as the search criterion. Alternatively, you can click **Advanced Search** and specify the search criterion.
   b. In the search results table on the left pane, click the scheduled job in the Job Name column.
4. On the Job Details tab, you can modify the parameters of the scheduled task:
   - **Retries**: Enter an integer value in this field. This number represents the number of times the scheduler tries to start the job before assigning the Stopped status to the job.

- **Schedule Type**: Depending on the frequency at which you want the job to run, select the appropriate schedule type. See Creating Jobs in *Oracle Fusion Middleware Administering Oracle Identity Governance*.

In addition to modifying the job details, you can enable or disable a job.

5. On the **Job Details** tab, in the Parameters region, specify values for the attributes of the scheduled task.

> **Note:**
>
> Values (either default or user-defined) must be assigned to all the attributes. If even a single attribute value is left empty, then reconciliation is not performed.

6. Click **Apply** to save the changes.

> **Note:**
>
> You can use the Scheduler Status page in Identity System Administration to either start, stop, or reinitialize the scheduler.

6.4 Guidelines on Performing Provisioning Operations

These are the guidelines that you must apply while performing provisioning operations.

- Guidelines on Performing Provisioning Operations for Oracle Database
- Guidelines on Performing Provisioning Operations for MySQL

6.4.1 Guidelines on Performing Provisioning Operations for Oracle Database

These are the guidelines that you must apply while performing provisioning operations.

- Before you perform provisioning operations, lookup definitions must be synchronized with the lookup fields of the target system. In other words, run the scheduled jobs for lookup field synchronization before provisioning operations.
- Passwords for user accounts provisioned from Oracle Identity Governance must adhere to the password policy set in the target system.
- The character length of target system fields must be taken into account when specifying values for the corresponding Oracle Identity Governance fields.
- During an update password provisioning operation, ensure that you clear the existing text in the Password field, and then enter the new password.
- During a Create User provisioning operation, the following are some of the fields that are optional:
  - Default Tablespace
  - Default Tablespace Quota (in MB)

    This field is dependent on Default Tablespace. To specify a quota, you must specify a value for Default Tablespace.

  - Temporary Tablespace
  - Profile Name

  If you specify a value for any of these fields during a Create User provisioning operation, then you must not leave them empty during an Update User provisioning operation. Otherwise, the provisioning operation will fail. However, you can modify the existing values in these fields.

- For creating password-authenticated database users, you must specify values for the following fields:
  - **Username:** Enter the name of the database user.
  - **Password:** Enter the password for the database user.
  - **Authentication Type:** Specify `PASSWORD` as the value of this lookup field.
- For creating globally-authenticated database users, you must specify a value for the following mandatory fields:
  - **Username:** Enter the name of the database user.
  - **Authentication Type:** Specify `GLOBAL` as the value of this lookup field.
  - **Global DN:** Enter the distinguished name (DN) for your organization.

    Sample value: `cn=ajones,cn=users,dc=oracle,dc=vm`

  After you submit the data required, the connector runs the following query to create a globally-authenticated database user:

  ```
  CREATE USER {__NAME__} IDENTIFIED GLOBALLY AS {globalDN}
  ```

- If you specify a value for the Default Tablespace Quota (in MB) field, then enter values in the following format:

  ```
  TABLESPACE_QUOTA M
  ```

  In this format, *TABLESPACE_QUOTA* is the tablespace quota allocated to the user and M indicates that megabytes is the unit of measurement of quota. The following is a sample value: `300 M`

  If you want to allocate to a user unlimited quota on a tablespace, then specify the following as the value of the Default Tablespace Quota (in MB) field:

  ```
  UNLIMITED
  ```

6.4.2 Guidelines on Performing Provisioning Operations for MySQL

These are the guidelines that you must apply while performing provisioning operations.

- Before you perform provisioning operations, lookup definitions must be synchronized with the lookup fields of the target system. In other words, run the scheduled jobs for lookup field synchronization before provisioning operations.
- Passwords for user accounts provisioned from Oracle Identity Governance must adhere to the password policy set in the target system.
- The character length of target system fields must be taken into account when specifying values for the corresponding Oracle Identity Governance fields.

- During an update password provisioning operation, ensure that you clear the existing text in the Password field, and then enter the new password.

6.5 Performing Provisioning Operations

You create a new user in Identity Self Service by using the Create User page. You provision or request for accounts on the Accounts tab of the User Details page.

To perform provisioning operations in Oracle Identity Governance:

1. Log in to Identity Self Service.
2. Create a user as follows:
   a. In Identity Self Service, click **Manage**. The Home tab displays the different Manage option. Click **Users**. The Manage Users page is displayed.
   b. From the Actions menu, select **Create**. Alternatively, you can click **Create** on the toolbar. The Create User page is displayed with input fields for user profile attributes.
   c. Enter details of the user in the Create User page.
3. On the Account tab, click **Request Accounts**.
4. In the Catalog page, search for and add to cart the application instance for the connector that you configured earlier, and then click **Checkout**.
5. Specify value for fields in the application form and then click **Ready to Submit**.
6. Click **Submit**.

> **See Also:**
>
> Creating a User in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance* for details about the fields on the Create User page

6.6 Uninstalling the Connector

Uninstalling the connector deletes all the account-related data associated with its resource objects.

If you want to uninstall the connector for any reason, then run the Uninstall Connector utility. Before you run this utility, ensure that you set values for `ObjectType` and `ObjectValues` properties in the ConnectorUninstall.properties file. For example, if you want to delete resource objects, scheduled tasks, and scheduled jobs associated with the connector, then enter `"ResourceObject", "ScheduleTask", "ScheduleJob"` as the value of the `ObjectType` property and a semicolon-separated list of object values corresponding to your connector (for example, `GoogleApps User; GoogleApps Group`) as the value of the `ObjectValues` property.

Note:

If you set values for the `ConnectorName` and `Release` properties along with the `ObjectType`and `ObjectValue` properties, then the deletion of objects listed in the `ObjectValues` property is performed by the utility and the Connector information is skipped.

For more information, see Uninstalling Connectors in *Oracle Fusion Middleware Administering Oracle Identity Governance*.

7
Extending the Database User Management Connector

You can extend the functionality of the connector to address your specific business requirements.

- Modifying the Predefined Queries or Creating New Queries
- Configuring Transformation and Validation of Data
- Configuring Action Scripts
- Configuring the Connector for Multiple Installations of the Target System

7.1 Modifying the Predefined Queries or Creating New Queries

Learn about the predefined queries for provisioning and reconciliation, and how to update them or create new ones to suit your requirements.

- Understanding the Predefined Queries for Oracle Database
- Understanding the Predefined Queries for MySQL
- Configuring Queries to Add Support for Custom Parameters and Lookup Fields

7.1.1 Understanding the Predefined Queries for Oracle Database

Learn about predefined queries, the syntax of the queries used for provisioning and reconciliation operations, the syntax of the list of values queries used for lookup field synchronization, and the guidelines that you must apply while modifying the predefined queries or creating new queries.

- About the Predefined Queries for Oracle Database
- Syntax of Provisioning Queries for Oracle Database
- Syntax of Reconciliation Queries for Oracle Database
- Syntax of List of Values Queries for Oracle Database
- Guidelines for Configuring Search Queries Used in Reconciliation from Oracle Databases

7.1.1.1 About the Predefined Queries for Oracle Database

The connector provides predefined SQL queries and stored procedures to reconcile target system user records, synchronize lookup field values with Oracle Identity Governance, and for provisioning operations. You can modify these predefined queries or add your own queries.

For example, to locate the reconciliation query file, you can extract the `/bundle/org.identityconnectors.dbum-1.0.1116.jar` file in the connector installation package and then open the `/scripts/oracle/Search.queries` file.

The connector includes the following types of queries:

- Provisioning Queries

  They are used for create, update, and delete operations. The query file is `scripts/oracle/Provisioning.queries`.

- List of Values Search Queries

  They are used for reconciliation of lookup definitions. A list of value query operates on a set of values for fields such as profiles, privileges, roles, and tablespaces. The query file is `scripts/oracle/LoVSearch.queries`.

- Account Search Queries

  They are used for full, incremental, and delete reconciliation operations. An account search query operates on account and group searches with various conditions. The query file is `scripts/oracle/Search.queries`.

  The following are the predefined queries for Oracle Database:

  – SEARCH_USER

    This query is used to fetch all user records from the DBA_USERS table.

  – BATCHED_SEARCH_USER

    This query is used to fetch from the DBA_USERS table user records that are present within the specified range. It is used to perform batched reconciliation on a target system that is configured as a target resource.

  – SEARCH_USER_ROLE

    This query is used to fetch all user roles from the DBA_ROLE_PRIVS table.

  – SEARCH_USER_PRIVILEGE

    This query is used to fetch all user privileges from the DBA_SYS_PRIVS table.

> **Note:**
>
> - The stored procedure OUT parameters cannot be configured for write-back on the process form. The returned values cannot be used for any connector operations.
> - Update operations for Oracle Database users are processed based on the create time-stamp, which is assigned to a user when the user is created. During incremental reconciliation, only the users created after this time-stamp are fetched. However, the users updated after the time-stamp are not fetched.

7.1.1.2 Syntax of Provisioning Queries for Oracle Database

The following is the syntax of the queries used for provisioning operations:

QUERYID {

Query="*QUERY*"

QueryType="*QUERYTYPE*"

Parameters=["*PARAM1*":"*PARAMDEFN1*", "*PARAM2*":"*PARAMDEFN2*"...]

ExtensionJoin="*EXTENSIONJOIN*"

ExtensionSeparator="*EXTENSIONSEPARATOR*"

QueryExtensions=["*EXTENSION1*","*EXTENSION2*"...]

}

For example:

```
CREATE_EXTERNAL_USER {
    Query="CREATE USER {__NAME__} IDENTIFIED EXTERNALLY"
    QueryType="SQL"
    Parameters=["__NAME__":"Type:String,TAGS:DOUBLEQUOTES"]
    ExtensionJoin=","
    ExtensionSeparator=", "
    QueryExtensions=["TEMP_TABLESPACE_QUERY","TABLESPACE_QUERY","PROFILE_QUERY"]
}
```

In this syntax:

- *QUERYID* refers to the unique name of the query.

  For example: CREATE_EXTERNAL_USER

  For CREATE provisioing queries, the format of *QUERYID* is CREATE_*AUTHENTICATIONTYPE_ACCOUNTTYPE.* The default account type is USER. For other provisioning queries, the format is the *OPERATIONTYPE_ATTRIBUTE*, such as UPDATE_GLOBALDN.
- *QUERY* refers to the main query.

  For example: Query="CREATE USER {__NAME__} IDENTIFIED EXTERNALLY"
- QueryType refers to the type of the main query, either an SQL query or a stored procedure. The value of *QUERYTYPE* can be SQL or StoredProc.

  For example: QueryType="SQL"
- Parameters refers to the list of comma separated parameters and parameter definitions used with the main query, represented by *"PARAM1":"PARAMDEFN1"*, "*PARAM2*":"*PARAMDEFN2"*, and so on.

  For example: Parameters=["__NAME__":"Type:String,TAGS:DOUBLEQUOTES"]

  A parameter can have the following attributes:

  - Type is the type of the parameter.
  - Direction is the flow of data from the query to or from the parameter. It can have a value of IN, OUT, or INOUT.
  - TAGS is the enclosure characters that are applied to each parameter before the query is processed. It can have a value of DOUBLEQUOTES, QUOTES, UPPERCASE, or LOWERCASE.

    If you want to use multiple tags, you must encapsulate the tags in escaped quotes and separate them by commas. However, you must not use DOUBLEQUOTES with QUOTES or UPPERCASE with LOWERCASE in the same query.

For example: `"Type:String,TAGS:\"DOUBLEQUOTES,UPPERCASE\"`

- ExtensionJoin (optional) refers to the operator, represented by *EXTENSIONJOIN,* used to join the main query with query extensions.

  For example: `ExtensionJoin=","`
- ExtensionSeparator (optional) refers to the delimiter between query extensions, represented by *EXTENSIONSEPARATOR.*

  For example: `ExtensionSeparator=", "`
- QueryExtensions (optional) refers to the extensions that must be appended to the main query, represented by *EXTENSION1, EXTENSION2,* and so on.

  For example:
  `QueryExtensions=["TEMP_TABLESPACE_QUERY","TABLESPACE_QUERY","PROFILE_QUERY"]`

During a provisioning operation, the connector combines all these components to the following query:

QUERY PARAM1, PARAM2... [*EXTENSIONJOIN* [*EXTENSION1 EXTENSIONSEPARATOR EXTENSION2 EXTENSIONSEPARATOR*...]]

For example:

```
CREATE USER {__NAME__} IDENTIFIED EXTERNALLY, TEMP_TABLESPACE_QUERY,
TABLESPACE_QUERY, PROFILE_QUERY
```

Table 7-1 lists the script selection logic of the provisioning queries:

Table 7-1 Script Section Logic for Oracle Provisioning Queries

Operation	Selection Logic	Query IDs
CREATE	CREATE_*AUTHTYPE_OBJECTYPE*	CREATE_PASSWORD_USER CREATE_GLOBAL_USER CREATE_EXTERNAL_USER
DELETE	DELETE_*OBJECTTTYPE*	DELETE_USER
ENABLE	ENABLE_*OBJECTTYPE*	ENABLE_USER
DISABLE	DISABLE_*OBJECTTYPE*	DISABLE_USER
RESET PASSWORD	SET_PASSWORD	SET_PASSWORD
UPDATE	UPDATE_*ATTRIBUTE*	UPDATE_TABLESPACE UPDATE_DEFAULTQUOTA UPDATE_GLOBALDN UPDATE_PROFILE UPDATE_TEMPTABLESPACE
ADD CHILD VALUES	UPDATE_ADD_*ATTRIBUTE*	UPDATE_ADD_ROLES UPDATE_ADD_PRIVILEGES
REMOVE CHILD VALUES	UPDATE_REVOKE_*ATTRIBUTE*	UPDATE_REVOKE_ROLES UPDATE_REVOKE_PRIVILEGES

7.1.1.3 Syntax of Reconciliation Queries for Oracle Database

The following is the syntax of the search queries used during reconciliation operations:

QUERYID {

Query="*QUERY*"

QueryType="*QUERYTYPE*"

Parameters=["*PARAM1*":"*PARAMDEFN1*", "*PARAM2*":"*PARAMDEFN2*"...]

ExtensionJoin="*EXTENSIONJOIN*"

ExtensionSeparator="*EXTENSIONSEPARATOR*"

QueryExtensions=["*EXTENSION1*","*EXTENSION2*"...]

}

For example:

```
SEARCH_USER {
    Query="SELECT {__UID__}, {authType}, {externalname}, {tablespace}, {status},
{tempTableSpace}, {profile}," +
        " {defaultQuota}, {tmpQuota}, {lastModified} FROM  DBA_USERS dba
{filter}"
    QueryType="SQL"
    Parameters=["__UID__":"Type:String,Direction:OUT,ColName:USERNAME",

"authType":"Type:String,Direction:OUT,ColName:PASSWORD,ColQuery:\"DECODE(PASSWORD
, 'EXTERNAL', 'EXTERNAL', 'GLOBAL', 'GLOBAL', 'PASSWORD')\"",
            "tablespace":"Type:String,Direction:OUT,ColName:DEFAULT_TABLESPACE",

"tmpQuota":"Type:String,Direction:OUT,ColName:TEMPORARY_TABLESPACE_QUOTA,ColQuery
:(SELECT MAX_BYTES FROM DBA_TS_QUOTAS WHERE dba.USERNAME = USERNAME AND
TABLESPACE_NAME = dba.TEMPORARY_TABLESPACE)",

"defaultQuota":"Type:String,Direction:OUT,ColName:DEFAULT_TABLESPACE_QUOTA,ColQue
ry:(SELECT MAX_BYTES FROM DBA_TS_QUOTAS WHERE dba.USERNAME = USERNAME AND
TABLESPACE_NAME = dba.DEFAULT_TABLESPACE)",
            "externalname":"Type:String,Direction:OUT,ColName:EXTERNAL_NAME",
            "status":"Type:String,Direction:OUT,ColName:ACCOUNT_STATUS",

"tempTableSpace":"Type:String,Direction:OUT,ColName:TEMPORARY_TABLESPACE",
            "profile":"Type:String,Direction:OUT,ColName:PROFILE",
            "lastModified":"Type:long,Direction:OUT,ColName:TIMESTAMP,
ColQuery:\"((CREATED - TO_DATE('01011970','ddmmyyyy')) *24*60*60*1000)\""]
    QueryExtensions=["SEARCH_USER_ROLE", "SEARCH_USER_PRIVILEGE"]
}
```

In this syntax:

- *QUERYID* refers to the unique name of the query.

  For example: `SEARCH_USER`

  QUERYID can be one of the following values:

  - `SEARCH_USER`
  - `BATCHED_SEARCH_USER`

  - SEARCH_USER_ROLE
  - SEARCH_USER_PRIVILEGE
- *QUERY* refers to the main query.

  For example: `Query="SELECT {__UID__}, {authType}, {externalname}, {tablespace}, {status}, {tempTableSpace}, {profile}," + " {defaultQuota}, {tmpQuota}, {lastModified} FROM DBA_USERS dba {filter}"`
- QueryType refers to the type of the main query, either an SQL query, a stored procedure, or a query extension. The value of *QUERYTYPE* can be `SQL`, `StoredProc`, or `QUERYEXTENSION`.

  For example: `QueryType="SQL"`
- Parameters refers to the list of comma separated parameters and parameter definitions used with the main query, represented by *"PARAM1":"PARAMDEFN1"*, "*PARAM2*":"*PARAMDEFN2"*, and so on.

  For example:

  `Parameters=["__UID__":"Type:String,Direction:OUT,ColName:USERNAME",`

  `"authType":"Type:String,Direction:OUT,ColName:PASSWORD,ColQuery:\"DECODE(PASSWORD, 'EXTERNAL', 'EXTERNAL', 'GLOBAL', 'GLOBAL', 'PASSWORD') \""]`

  A parameter can have the following attributes:
  - Type is the type of the parameter.
  - Direction is the flow of data from the query to or from the parameter. It can have a value of `IN`, `OUT`, or `INOUT`.
  - ColName is the column name in the target system corresponding to the parameter in the query.
  - ColQuery is the query used to fetch values for the corresponding query parameter.
- ExtensionJoin (optional) refers to the operator, represented by *EXTENSIONJOIN,* used to join the main query with query extensions.

  For example: `ExtensionJoin=","`
- ExtensionSeparator (optional) refers to the delimiter between query extensions, represented by *EXTENSIONSEPARATOR.*

  For example: `ExtensionSeparator=", "`
- QueryExtensions (optional) refers to the extensions that must be appended to the main query, represented by *EXTENSION1, EXTENSION2,* and so on.

  For example: `QueryExtensions=["SEARCH_USER_ROLE", "SEARCH_USER_PRIVILEGE"]`

During a reconciliation operation, the connector combines all these components to the following query:

QUERY PARAM1, PARAM2... [*EXTENSIONJOIN* [*EXTENSION1 EXTENSIONSEPARATOR EXTENSION2 EXTENSIONSEPARATOR*...]]

For example:

```
SELECT {__UID__}, {authType}, {externalname}, {tablespace}, {status},
{tempTableSpace}, {profile}, {defaultQuota}, {tmpQuota}, {lastModified}
FROM DBA_USERS dba {filter}, SEARCH_USER_ROLE, SEARCH_USER_PRIVILEGE
```

7.1.1.4 Syntax of List of Values Queries for Oracle Database

If a search query is performed on account types, such as User Name, then the query is considered as a reconciliation query. If a search query is performed on any other object, then the query is considered as a list of values query.

The following is the syntax of the list of values queries used for lookup field synchronization:

OBJECTTYPE = "*QUERY*"

For example:

```
__PROFILE__="SELECT DISTINCT profile FROM dba_profiles"
```

In this syntax:

- *OBJECTTYPE* refers to the lookup field attribute.

  For example: `__PROFILE__`

- *QUERY* refers to the query used for fetching a lookup field attribute.

  For example: `SELECT DISTINCT profile FROM dba_profiles`

The list of values queries return values that are used as lookup field entries. By default, the connector includes dedicated scheduled job for each lookup definition. To use a custom lookup definition, you must add custom fields in the query file.

7.1.1.5 Guidelines for Configuring Search Queries Used in Reconciliation from Oracle Databases

The following are guidelines that you must apply while modifying or creating queries for reconciliation:

- By adding or removing a column from the SELECT clause of a reconciliation query, you add or remove an attribute from the list of target system attributes for reconciliation. To enable the connector to process a change (addition or removal) in the list of reconciled attributes, you must make corresponding changes in the provisioning part of the connector.

  If there are any read-only attributes, then you must disable updates to the read-only attributes in the respective process forms.

- In the query properties file, you must not change the names of the predefined queries.
- Some of the predefined queries use inner queries. If you add or remove a column from the outer query, you must make corresponding changes in the inner queries.
- You cannot remove columns corresponding to the User Name resource object attribute.
- You must ensure that the following condition included in the Parameters list is not removed:

```
"lastModified":"Type:long,Direction:IN,ColQuery:\"((CREATED -
TO_DATE('01011970','ddmmyyyy')) *24*60*60*1000)\""]
```

This condition is used to determine if a target system record was added or updated after the time-stamp stored in the Incremental Recon Attribute scheduled job attribute.

- You must ensure that formats for date literals are specified by the use of the TO_DATE function. For example, instead of specifying a date value as `'31-Dec-4712'` use `TO_DATE('31-Dec-4712','DD-Mon-YYYY')`.
- When you add or remove columns from the SELECT clause of the queries in the properties file, then you must update the attribute mapping lookup definition that holds mappings between child attributes and the target system column names. In addition, you must update other OIM objects.
- Before you modify or add a query in the Search.queries file, you must run the query by using any standard database client to ensure that the query produces the required results when it is run against the target system database.

7.1.2 Understanding the Predefined Queries for MySQL

Learn about predefined queries, the syntax of the queries used for provisioning and reconciliation operations, and the syntax of the list of values queries used for lookup field synchronization.

- About the Queries for MySQL Database
- Syntax of Provisioning Queries for MySQL Database
- Syntax of Reconciliation Queries for MySQL Database
- Syntax of List of Values Queries for MySQL Database

7.1.2.1 About the Queries for MySQL Database

The connector provides predefined SQL queries and stored procedures to reconcile target system user records, synchronize lookup field values with Oracle Identity Governance, and for provisioning operations. You can modify these predefined queries or add your own queries.

For example, to locate the reconciliation query file, you can extract the/ `bundle/org.identityconnectors.dbum-1.0.1116.jar` file in the connector installation package and then open the `/scripts/mysql/Search.queries` file.

The connector includes the following types of queries:

- Provisioning Queries

  They are used for create, update, and delete operations. The query file is / `scripts/mysql/Provisioning.queries`.
- List of Values Search Queries

  They are used for reconciliation of lookup definitions. A list of value query operates on a set of values for fields such as profiles, privileges, roles, and tablespaces. The query file is `/scripts/mysql/LoVSearch.queries`.
- Account Search Queries

They are used for full and delete reconciliation operations. An account search query operates on account and group searches with various conditions. The query file is `/scripts/mysql/Search.queries`.

The following are the predefined queries for Oracle Database:

- SEARCH_USER

  This query is used to fetch all user records from the mysql.user table.
- BATCHED_SEARCH_USER

  This query is used to fetch from the mysql.user table user records that are present within the specified range. It is used to perform batched reconciliation on a target system that is configured as a target resource.
- SEARCH_USER_ROLE

  This query is used to fetch all user roles from the information_schema.SCHEMA_PRIVILEGES table.
- SEARCH_USER_PRIVILEGE

  This query is used to fetch all user privileges from the DBA_SYS_PRIVS table.

> **Note:**
>
> The stored procedure OUT parameters cannot be configured for write-back on the process form. The returned values cannot be used for any connector operations.

7.1.2.2 Syntax of Provisioning Queries for MySQL Database

The following is the syntax of the queries used for provisioning operations:

QUERYID {

Query="*QUERY*"

QueryType="*QUERYTYPE*"

Parameters=["*PARAM1*":"*PARAMDEFN1*", "*PARAM2*":"*PARAMDEFN2*"...]

ExtensionJoin="*EXTENSIONJOIN*"

ExtensionSeparator="*EXTENSIONSEPARATOR*"

QueryExtensions=["*EXTENSION1*","*EXTENSION2*"...]

}

For example:

```
CREATE_USER {
    Query="CREATE USER {__NAME__} IDENTIFIED BY {__PASSWORD__}"
    QueryType="SQL"

Parameters=["__NAME__":"Type:String","__PASSWORD__":"Type:GuardedString,TAGS:QUOT
ES"]
    QueryExtensions=[]
}
```

In this syntax:

- *QUERYID* refers to the unique name of the query.

  For example: `CREATE_USER`
- *QUERY* refers to the main query.

  For example: `Query="CREATE USER {__NAME__} IDENTIFIED BY {__PASSWORD__}"`
- QueryType refers to the type of the main query, either an SQL query or a stored procedure. The value of *QUERYTYPE* can be `SQL` or `StoredProc`.

  For example: `QueryType="SQL"`
- Parameters refers to the list of comma separated parameters and parameter definitions used with the main query, represented by *"PARAM1":"PARAMDEFN1"*, "*PARAM2*":"*PARAMDEFN2*", and so on.

  For example: `Parameters=["__NAME__":"Type:String","__PASSWORD__":"Type:GuardedString,TAGS:QUOTES"]`

  A parameter can have the following attributes:

  - Type is the type of the parameter.
  - Direction is the flow of data from the query to or from the parameter. It can have a value of `IN`, `OUT`, or `INOUT`.
  - TAGS is the enclosure characters that are applied to each parameter before the query is processed. It can have a value of `DOUBLEQUOTES`, `QUOTES`, `UPPERCASE`, or `LOWERCASE`.

    If you want to use multiple tags, you must encapsulate the tags in escaped quotes and separate them by commas. However, you must not use `DOUBLEQUOTES` with `QUOTES` or `UPPERCASE` with `LOWERCASE` in the same query.

    For example: `"Type:String,TAGS:\"DOUBLEQUOTES,UPPERCASE\"`
- ExtensionJoin (optional) refers to the operator, represented by *EXTENSIONJOIN,* used to join the main query with query extensions.

  For example: `ExtensionJoin=","`
- ExtensionSeparator (optional) refers to the delimiter between query extensions, represented by *EXTENSIONSEPARATOR.*

  For example: `ExtensionSeparator=", "`
- QueryExtensions (optional) refers to the extensions that must be appended to the main query, represented by *EXTENSION1, EXTENSION2,* and so on.

During a provisioning operation, the connector combines all these components to the following query:

QUERY PARAM1, PARAM2... [*EXTENSIONJOIN* [*EXTENSION1 EXTENSIONSEPARATOR EXTENSION2 EXTENSIONSEPARATOR*...]]

For example:

```
CREATE USER {__NAME__} IDENTIFIED BY {__PASSWORD__}
```

Table 7-2 lists the script selection logic of the provisioning queries:

Table 7-2 Script Section Logic for MySQL Provisioning Queries

Operation	Selection Logic	Query IDs
CREATE	CREATE_*OBJECTYPE*	CREATE_USER
DELETE	DELETE_*OBJECTTTYPE*	DELETE_USER
RESET PASSWORD	SET_PASSWORD	SET_PASSWORD
ADD CHILD VALUES	UPDATE_ADD_*ATTRIBUTE*	UPDATE_ADD_PRIVILEGES
REMOVE CHILD VALUES	UPDATE_REVOKE_*ATTRIBUTE*	UPDATE_REVOKE_PRIVILEGES

7.1.2.3 Syntax of Reconciliation Queries for MySQL Database

The following is the syntax of the search queries used during reconciliation operations:

QUERYID {

Query="*QUERY*"

QueryType="*QUERYTYPE*"

Parameters=["*PARAM1*":"*PARAMDEFN1*", "*PARAM2*":"*PARAMDEFN2*"...]

ExtensionJoin="*EXTENSIONJOIN*"

ExtensionSeparator="*EXTENSIONSEPARATOR*"

QueryExtensions=["*EXTENSION1*","*EXTENSION2*"...]

}

For example:

```
SEARCH_USER {
    Query="SELECT {__UID__} FROM MYSQL.USER {filter}"
    QueryType="SQL"
    Parameters=["__UID__":"Type:String,Direction:OUT,ColName:USER"]
    QueryExtensions=["SEARCH_USER_PRIVILEGE"]
}
```

In this syntax:

- *QUERYID* refers to the unique name of the query.

  For example: `SEARCH_USER`

  QUERYID can be one of the following values:

  - `SEARCH_USER`
  - `BATCHED_SEARCH_USER`
  - `SEARCH_USER_PRIVILEGE`

- *QUERY* refers to the main query.

  For example: `Query="SELECT {__UID__} FROM MYSQL.USER {filter}"`

- QueryType refers to the type of the main query, either an SQL query, a stored procedure, or a query extension. The value of *QUERYTYPE* can be `SQL`, `StoredProc`, or `QUERYEXTENSION`.

  For example: `QueryType="SQL"`
- Parameters refers to the list of comma separated parameters and parameter definitions used with the main query, represented by *"PARAM1":"PARAMDEFN1"*, "*PARAM2*":"*PARAMDEFN2"*, and so on.

  For example:

  `Parameters=["__UID__":"Type:String,Direction:OUT,ColName:USER"]`

  A parameter can have the following attributes:
  - Type is the type of the parameter.
  - Direction is the flow of data from the query to or from the parameter. It can have a value of `IN`, `OUT`, or `INOUT`.
  - ColName is the column name in the target system corresponding to the parameter in the query.
  - ColQuery is the query used to fetch values for the corresponding query parameter.
- ExtensionJoin (optional) refers to the operator, represented by *EXTENSIONJOIN*, used to join the main query with query extensions.

  For example: `ExtensionJoin=","`
- ExtensionSeparator (optional) refers to the delimiter between query extensions, represented by *EXTENSIONSEPARATOR*.

  For example: `ExtensionSeparator=", "`
- QueryExtensions (optional) refers to the extensions that must be appended to the main query, represented by *EXTENSION1, EXTENSION2,* and so on.

  For example: `QueryExtensions=["SEARCH_USER_PRIVILEGE"]`

During a reconciliation operation, the connector combines all these components to the following query:

QUERY PARAM1, PARAM2... [*EXTENSIONJOIN* [*EXTENSION1 EXTENSIONSEPARATOR EXTENSION2 EXTENSIONSEPARATOR*...]]

For example:

```
SELECT {__UID__} FROM MYSQL.USER {filter} SEARCH_USER_PRIVILEGE
```

7.1.2.4 Syntax of List of Values Queries for MySQL Database

If a search query is performed on account types, such as User Name, then the query is considered as a reconciliation query. If a search query is performed on any other object, then the query is considered as a list of values query.

The following is the syntax of the list of values queries used for lookup field synchronization:

OBJECTTYPE = "*QUERY*"

For example:

```
__PRIVILEGES__="SELECT CONCAT(p.PRIVILEGE_TYPE, ' ON
',s.SCHEMA_NAME) SchemaPrivilege FROM INFORMATION_SCHEMA.SCHEMATA
s,INFORMATION_SCHEMA.SCHEMA_PRIVILEGES p"
```

In this syntax:

- *OBJECTTYPE* refers to the lookup field attribute.

  For example: `__PRIVILEGES__`
- *QUERY* refers to the query used for fetching a lookup field attribute.

  For example: `SELECT CONCAT(p.PRIVILEGE_TYPE, ' ON ',s.SCHEMA_NAME) SchemaPrivilege FROM INFORMATION_SCHEMA.SCHEMATA s,INFORMATION_SCHEMA.SCHEMA_PRIVILEGES p`

The list of values queries return values that are used as lookup field entries. By default, the connector includes dedicated scheduled job for each lookup definition. To use a custom lookup definition, you must add custom fields in the query file.

7.1.3 Configuring Queries to Add Support for Custom Parameters and Lookup Fields

The connector uses preconfigured queries for connector operations such as create, delete, and search. You can add custom parameters and lookup definition fields as per your requirements.

The following sections provide the procedure to add a parameter or a lookup definition field to a query file:

- Updating the Query Files
- Configuring Oracle Identity Governance for Custom Parameters

7.1.3.1 Updating the Query Files

To update the query files for Oracle Database or MySQL:

1. Run the Oracle Identity Governance Download JARs utility to download the connector bundle JAR file from the Oracle Identity Governance database. This utility is copied into the following location when you install Oracle Identity Governance:

   > **Note:**
   >
   > Before you use this utility, verify that the `WL_HOME` environment variable is set to the directory in which Oracle WebLogic Server is installed.

   For Microsoft Windows:

   OIM_HOME/server/bin/DownloadJars.bat

   For UNIX:

   OIM_HOME/server/bin/DownloadJars.sh

When you run the utility, you are prompted to enter the login credentials of the Oracle Identity Governance administrator, URL of the Oracle Identity Governance host computer, context factory value, type of JAR file being downloaded, and the location from which the JAR file is to be downloaded. Select ICFBundle as the JAR type.

2. Copy the bundle JAR file in a temporary directory.

   Sample JAR file: `bundle/org.identityconnectors.dbum-12.3.0.jar`

   Sample temporary directory: `c:\temp`

3. Run the following command to extract the connector bundle JAR file:

   ```
   jar -xvf org.identityconnectors.dbum-12.3.0.jar
   ```

   > **Note:**
   >
   > You can also run the WinZip or WinRAR utility to extract the contents from the JAR file.

4. Delete the bundle JAR file in the temporary directory.

5. Depending on your requirement, update the query files with new parameters as per the query syntax.

   - For Oracle Database:

     For example, if you want to add a new parameter, tmpQuota, to the CREATE_USER provisioning query, then:

     a. Open the provisioning query file in a text editor.

        Sample query file: `c:\temp\bundle\org.identityconnectors.dbum-12.3.0\scripts\oracle\Provisioning.queries`

     b. Add the parameter, `tmpQuota`, to the `CREATE_USER` query.

        The following is a sample updated query:

        ```
        CREATE_USER {
            Query="CREATE USER {__NAME__} IDENTIFIED BY
        {__PASSWORD__} TEMPORARY QUOTA {tmpQuota} ON
        {tempTableSpace}"
            QueryType="SQL"
            Parameters=["__NAME__":"Type:String,TAGS:DOUBLEQUOTES",
        "__PASSWORD__":"Type:GuardedString,TAGS:DOUBLEQUOTES",
        "tmpQuota":"Type:String",
        "tempTableSpace":"Type:String,Tags:EXCLUDE_VALIDATION"]

        QueryExtensions=["TABLESPACE_QUERY","TEMP_TABLESPACE_QUERY","
        PROFILE_QUERY","DEFAULTS_QUOTA_QUERY","TEMPTS_QUOTA_QUERY"]
        }
        ```

     c. Save and close the query file.

   - For MySQL:

For example, if you want to add a new parameter, CUSTOM_ATTRIBUTE, to the CREATE_USER provisioning query, then:

a. Open the provisioning query file in a text editor.

   Sample query file:
   `c:\temp\bundle\org.identityconnectors.dbum-12.3.0\scripts\mysql\Provisioning.queries`

b. Add the parameter, `CUSTOM_ATTRIBUTE`, to the `CREATE_USER` query.

   The following is a sample updated query:

```
CREATE_USER {
    Query="CREATE USER {__NAME__} IDENTIFIED BY
{__PASSWORD__}, {CUSTOM_ATTRIBUTE}"
    QueryType="SQL"
    Parameters=["__NAME__":"Type:String",
"__PASSWORD__":"Type:GuardedString,TAGS:QUOTES",
"CUSTOM_ATTRIBUTE":"Type:String,Direction:IN"]
    QueryExtensions=[]
}
```

c. Save and close the query file.

6. Create a new bundle JAR file that contains the updated manifest file and the provisioning query file as follows:

   a. Open the command prompt and navigate to the temporary directory `c:\temp`.

   b. Regenerate the connector bundle (org.identityconnectors.dbum-12.3.0.jar) by running the following command:

```
jar -cvfm org.identityconnectors.dbum-12.3.0.jar META-INF/
MANIFEST.MF *
```

> **Note:**
>
> While updating the connector bundle, ensure that META-INF\MANIFEST.MF file is unchanged.

7. In the case of a remote connector server, copy the new bundle JAR file in the bundles directory of the remote connector server, instead of posting the JAR file to the Oracle Identity Governance database.

8. Run the Oracle Identity Governance Upload JARs utility to upload the regenerated connector bundle to Oracle Identity Governance database. This utility is copied into the following location when you install Oracle Identity Governance:

> **Note:**
>
> Before you use this utility, verify that the `WL_HOME` environment variable is set to the directory in which Oracle WebLogic Server is installed.

- For Microsoft Windows:

  OIM_HOME/server/bin/UploadJars.bat

- For UNIX:

  OIM_HOME/server/bin/UploadJars.sh

When you run the utility, you are prompted to enter the login credentials of the Oracle Identity Governance administrator, URL of the Oracle Identity Governance host computer, context factory value, type of JAR file being uploaded, and the location from which the JAR file is to be uploaded. Specify 4 (ICF Bundle) as the value of the JAR type.

> **See Also:**
>
> Understanding the Predefined Queries for Oracle Database or Understanding the Predefined Queries for MySQLfor information about the syntax of the queries that you need to update

7.1.3.2 Configuring Oracle Identity Governance for Custom Parameters

Add the custom parameters that you added to the query files to the Schema form in Oracle Identity Governance.

> **Note:**
>
> Skip this procedure if the parameter you added already exists as a default form field in Oracle Identity Governance.

To add the parameters that you added to the query files to the Schema form in Identity Self Service, see Providing Schema Information for Target Application or Providing Schema Information for Authoritative Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

For example, if you are using Oracle Database as the target system and you added the `tmpQuota` parameter to the `CREATE_USER` provisioning query, then update the Schema form corresponding to your application in Identity Self Service to include details of the newly added parameter. The following are some sample values:

- Display Name: Temporary Quota
- Target Name: tmpQuota
- DataType: String
- Provisioning Field?: Yes

7.2 Configuring Transformation and Validation of Data

Configure transformation and validation of user account data by writing Groovy script logic while creating your application.

You can configure transformation of reconciled single-valued user data according to your requirements. For example, you can use First Name and Last Name values to create a value for the Full Name field in Oracle Identity Governance.

Similarly, you can configure validation of reconciled and provisioned single-valued data according to your requirements. For example, you can validate data fetched from the First Name attribute to ensure that it does not contain the number sign (#). In addition, you can validate data entered in the First Name field on the process form so that the number sign (#) is not sent to the target system during provisioning operations.

To configure transformation or validation of user account data, you must write Groovy scripts while creating your application. For more information about writing Groovy script-based validation and transformation logic, see Validation and Transformation of Provisioning and Reconciliation Attributes of *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

7.3 Configuring Action Scripts

You can configure **Action Scripts** by writing your own Groovy scripts while creating your application.

These scripts can be configured to run before or after the create, update, or delete an account provisioning operations. For example, you can configure a script to run before every user creation operation.

For information on adding or editing action scripts, see Updating the Provisioning Configuration in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

7.4 Configuring the Connector for Multiple Installations of the Target System

You must create copies of configurations of your base application to configure it for multiple installations of the target system.

The following example illustrates this requirement:
The London and New York offices of Example Multinational Inc. have their own installations of the target system, including independent schema for each. The company has recently installed Oracle Identity Governance, and they want to configure it to link all the installations of the target system.

To meet the requirement posed by such a scenario, you must clone your application which copies all configurations of the base application into the cloned application. For more information about cloning applications, see Cloning Applications in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

8

Upgrading the Database User Management Connector

If you have already deployed the 11.1.1.8.0 version of the Database User Management connector, then you can upgrade the connector to version 12.2.1.3.0 by uploading the new connector JAR files to the Oracle Identity Manager database.

Note:

- If you have deployed the 11.1.1.6.0 or earlier version of the Database User Management connector, you must first upgrade the connector to version 11.1.1.8.0. See "Upgrading the Connector" in Oracle Identity Manager Connector Guide for Database User Management.
- Before you perform the upgrade procedure:
  - It is strongly recommended that you create a backup of the Oracle Identity Manager database. Refer to the database documentation for information about creating a backup.
  - As a best practice, perform the upgrade procedure in a test environment initially.

The following sections discuss the procedure to upgrade the connector:

- Upgrade Steps
- Postupgrade Steps

See Also:

Upgrading Connectors in *Oracle Fusion Middleware Administering Oracle Identity Manager* for detailed information on these steps

8.1 Upgrade Steps

This is a summary of the procedure to upgrade the connector for both staging and production environments.

Depending on the environment in which you are upgrading the connector, perform one of the following steps:

- Staging Environment

  Perform the upgrade procedure by using the wizard mode.

> **Note:**
>
> Do not upgrade IT resource type definition. In order to retain the default setting, you must map the IT resource definition to "None".

- Production Environment

  Perform the upgrade procedure by using the silent mode.

8.2 Postupgrade Steps

Postupgrade steps involve uploading new connector jars, configuring the upgraded IT resource of the source connector, deploying the Connector Server, and configuring the latest token value of the scheduled job.

Perform the following procedure:

1. Upload new connector JARs as:
   a. Run the Upload JARs utility ($*ORACLE_HOME*/bin/UploadJars.sh) for uploading connector JARs.
   b. Upload bundle/org.identityconnectors.dbum-12.3.0.jar as ICFBundle.

   > **Note:**
   >
   > If you need to add a third-party JAR:
   >
   > - Navigate to the bundle directory.
   > - Create /lib folder and drop the third party jar in that folder.
   > - Update the bundle with library "jar uvf org.identityconnectors.dbum-12.3.0.jar lib/*FILE_NAME*".

   c. Upload lib/DBUM-oim-integration.jar as JavaTask.
2. Replicate all changes made to the Form Designer of the Design Console in a new UI form as follows:
   a. Log in to Oracle Identity System Administration.
   b. Create and activate a sandbox.
   c. Create a new UI form to view the upgraded fields.
   d. Associate the newly created UI form with the application instance of your target system. To do so, open the existing application instance for your resource, from the Form field, select the form (created in step 2.c) and then save the application instance.
   e. Publish the sandbox.
3. Configure the upgraded IT resource of the source connector.
4. Deploy the Connector Server.
5. Configure the latest token value of the scheduled job as follows:

The following scheduled jobs contain the Latest Token attribute:

For Oracle

- DBUM Oracle User Target Reconciliation
- DBUM Oracle User Trusted Reconciliation

For MSSQL:

- DBUM MSSQL Trusted Reconciliation
- DBUM MSSQL User Login Target Reconciliation
- DBUM MSSQL User Target Reconciliation

After upgrading the connector, you can perform either full reconciliation or incremental reconciliation. This ensures that records created or modified since the last reconciliation run are fetched into Oracle Identity Manager. From the next reconciliation run onward, the reconciliation engine automatically enters a value for the Latest Token attribute.

Note:

If there are customizations in the query files, to include custom parameters, and for transformation/validation of data during reconciliation/ provisioning, then the same customizations have to be performed in the respective query files after upgrading the connector.

See Also:

- Configuring Oracle Identity Governance for information about creating, activating, and publishing a sandbox, and creating a new UI form.
- Using an Identity Connector Server in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager* for information about deploying the Connector Server.
- Configuring Reconciliation for Oracle Database and Configuring Reconciliation for MySQL for more information about performing full or incremental reconciliation for Oracle and MySQL databases respectively.

A

Files and Directories in the Database User Management Connector Installation Package

These are the components of the connector installation package that comprise the Database User Management connector.

Table A-1 lists the files and directories on the connector installation package that comprise the Database User Management connector.

Table A-1   Files and Directories in the Database User Management Connector Installation Package

File in the Installation Package	Description
bundle/org.identityconnectors.dbum-12.3.0.jar	This file contains connector code, SQL queries, and stored procedures that are used for provisioning and reconciliation.
Files in the configuration directory: DBUM-DB2-CI.xml DBUM-MSSQL-CI.xml DBUM-MySQL-CI.xml DBUM-Oracle-CI.xml DBUM-Sybase-CI.xml	These files are used for installing a CI-based connector. This directory contains the configuration files that are used by the Connector Installer during installation of the connector for a particular target system.
Files in the javadoc directory	This directory contains information about the Java APIs used by the connector.
lib/DBUM-oim-integration.jar	This JAR file contains the class files that are used during reconciliation and provisioning operations. During connector deployment, this file is copied to the Oracle Identity Governance database.
Files in the resources directory	Each of these resource bundles contains language-specific information that is used by the connector. During connector deployment, this file is copied to the Oracle Identity Governance database. **Note:** A **resource bundle** is a file containing localized versions of the text strings that include GUI element labels and messages.
Files in the upgrade directory: PostUpgradeScriptOracleDBUM.sql PostUpgradeScriptMSSQLDBUM.sql PostUpgradeScriptMySQLDBUM.sql PostUpgradeScriptDB2DBUM.sql PostUpgradeScriptSybaseDBUM.sql	This directory contains the scripts for performing the postupgrade operations.

Table A-1 (Cont.) Files and Directories in the Database User Management Connector Installation Package

File in the Installation Package	Description
xml/DBUserManagement-MySQL-target-template.xml xml/DBUserManagement-Oracle-target-template.xml	This file contains definitions for the connector objects required for creating a Target application. It includes certain details required to connect Oracle Identity Governance with the target system . It also includes configuration details specific to your target system, attribute mappings, correlation rules, and reconciliation jobs.
xml/DBUserManagement-MySQL-auth-template.xml xml/DBUserManagement-Oracle-auth-template.xml	This file contains definitions for the connector objects required for creating an Authoritative application. It includes certain details required to connect Oracle Identity Governance with the target system . It also includes configuration details specific to your target system, attribute mappings, correlation rules, and reconciliation jobs.
xml/DBUserManagement-MySQL-pre-config.xml xml/DBUserManagement-Oracle-pre-config.xml	This XML file contains definitions for the connector objects associated with any non-User objects such as Roles and Privileges.
ConnectorConfig and Datasets files in the xml directory: DBUserManagement-Oracle-ConnectorConfig.xml DBUserManagement-Oracle-Datasets.xml DBUserManagement-MSSQL-ConnectorConfig.xml DBUserManagement-MSSQL-Datasets.xml DBUserManagement-MySQL-ConnectorConfig.xml DBUserManagement-MySQL-Datasets.xml DBUserManagement-DB2-ConnectorConfig.xml DBUserManagement-DB2-Datasets.xml DBUserManagement-Sybase-ConnectorConfig.xml DBUserManagement-Sybase-Datasets.xml **Note:** The dataset XML files are applicable only if you are using Oracle Identity Manager release 11.1.1.*x*.	These are the configuration (target and trusted) XML files and dataset XML files specific to the target system. The configuration XML files contain definitions for the various connector objects, such as resource objects and scheduled jobs, where as the dataset XML files contain datasets for the request based operations. • IT resource type • Process form for each login entity • Process form for each user entity • Process tasks for each login entity • Process tasks for each user entity • Resource objects for each login entity • Resource objects for each user entity • Provisioning Processes for each login entity **Note:** These files are applicable only for a CI-based connector.

Index