Oracle® Identity Manager

Connector Guide for Google Apps

Release 11.1.1
E18798-18
May 2020

Oracle Identity Manager Connector Guide for Google Apps, Release 11.1.1

E18798-18

Primary Author: Gowri G.R

Contributing Authors: Alankrita Prakash

Contents

2 Deploying the Google Apps Connector

3 Using the Connector

4 Extending the Functionality of the Connector

List of Figures

List of Tables

Preface

This guide describes the connector that is used to onboard Google Apps applications to 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:

http://docs.oracle.com/middleware/12213/oig/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:

http://docs.oracle.com/middleware/oig-connectors-12213/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 Oracle Identity Manager Connector for Google Apps?

These are the updates made to the software and documentation for the Oracle Identity Manager Connector for Google Apps in release 11.1.1.7.0.

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

- Software Updates

  This section describes updates made to the connector software.

- Documentation-Specific Updates

  This section describes major changes made to this guide. These changes are not related to software updates.

Software Updates

These are the updates made to the connector software.

The following section discusses software updates:

- Software Updates in Release 11.1.1.7.0
- Software Updates in Release 11.1.1.6.0
- Software Updates in Release 11.1.1.5.0

Software Updates in Release 11.1.1.7.0

These are the updates made to Release 11.1.1.7.0 of the connector.

The following are the software updates in release 11.1.1.7.0:

- Connector Redesigned to Use the New Directory API
- Support for Oracle Identity Manager 11*g* Release 2 PS2
- Support for Batched Reconciliation
- Support for Group Provisioning and Reconciliation
- Support for Provisioning Users to Organizations
- Support for Connector Operations in Multiple Domains

Connector Redesigned to Use the New Directory API

In the earlier releases, this connector used the Google Apps Provisioning APIs to connect to the target system and perform connector operations. From this release onward, the connector uses the Admin SDK's Directory API.

Support for Oracle Identity Manager 11*g* Release 2 PS2

From this release onward, the connector can be installed and used on 11*g* Release 2 PS2 (11.1.2.2.0) and any later BP in this release track. This Oracle Identity Manager version is mentioned in Certified Components.

Support for Batched Reconciliation

From this release onward, this connector supports batched reconciliation. See Batched Reconciliation for more information.

Support for Group Provisioning and Reconciliation

From this release onward, this connector supports provisioning and reconciliation of groups. See the following sections for more information:

- Lookup.GoogleApps.GM.Configuration
- GoogleApps Group Recon
- GoogleApps Group Delete Recon

Support for Provisioning Users to Organizations

From this release onward, you can provision users to specific organizations. The Org Unit Path field has been added to the OIM User form to enable provisioning of users to organizations.

Support for Connector Operations in Multiple Domains

The connector can be configured to perform connector operations in more than one domain by specifying a value for the supportMultipleDomain entry of the Lookup.Configuration.GoogleApps lookup definition.

See the following sections for more information:

- Support for Connector Operations in Multiple Domains
- Lookup.Configuration.GoogleApps

Software Updates in Release 11.1.1.6.0

These are the updates made to Release 11.1.1.6.0 of the connector.

The following are issues resolved in release 11.1.1.6.0:

Bug Number	Issue	Resolution
12380539	Reconciliation event of the emptied target child table was ignored.	This issue has been resolved.
12420420	WebLogic logging configuration was ignored.	This issue has been resolved.

Bug Number	Issue	Resolution
16530509	There was no tagging of the AccountName, AccountID, Entitlement and ITResource properties for the process form fields of the target systems.	This issue has been resolved. The AccountName, AccountID, Entitlement and ITResource properties of the process form fields have been tagged.

Software Updates in Release 11.1.1.5.0

These are the updates made to Release 11.1.1.5.0 of the connector.

This is the first release of the Oracle Identity Manager connector for GoogleApps based on ICF architecture. Therefore, there are no software updates in this release.

Documentation-Specific Updates

These are the updates made to the connector documentation.

The following section discusses documentation-specific updates:

- Documentation-Specific Updates in Release 11.1.1.7.0
- Documentation-Specific Updates in Release 11.1.1.6.0
- Documentation-Specific Updates in Release 11.1.1.5.0

Documentation-Specific Updates in Release 11.1.1.7.0

These are the updates made to Release 11.1.1.7.0 of the connector.

The following documentation-specific update has been made in revision "18" of release 11.1.1.7.0:

The "Oracle Identity Governance or Oracle Identity Manager" row of Certified Components has been updated to include support for Oracle Identity Governance release 12*c* PS4 (12.2.1.4.0).

The following documentation-specific update has been made in revision "17" of release 11.1.1.7.0:

The "Oracle Identity Manager" row of Certified Components has been renamed as "Oracle Identity Governance or Oracle Identity Manager" and also updated for Oracle Identity Governance 12*c* (12.2.1.3.0) certification.

The following documentation-specific update has been made in revision "16" of release 11.1.1.7.0:

- The "JDK" row has been added to Certified Components.
- Oracle Identity Manager interface names have been corrected throughout the document.
- Preinstallation on the Target System has been modified to include a high-level summary of steps to be performed on the target system. As a result, the following sections have been removed:
  - Registering the Client Application

  - Activating API Services
  - Creating a Service Account
  - Adding Scopes for the Client Application
  - Creating a Target System User Account for Connector Operations
  - Enabling Administrative API for Google Apps Business Domain

The following documentation-specific update has been made in revision "15" of release 11.1.1.7.0:

Upgrading the Connector has been updated.

The following documentation-specific update has been made in revision "14" of release 11.1.1.7.0:

The "Oracle Identity Manager" row of Certified Components has been updated.

The following documentation-specific update has been made in revision "13" of release 11.1.1.7.0:

A "Note" has been added at the beginning of Extending the Functionality of the Connector.

The following documentation-specific update has been made in revision "12" of release 11.1.1.7.0:

A screenshot in Section 2.1.2.4, "Adding Scopes for the Client Application" has been modified.

The following documentation-specific updates have been made in revision "11" of the guide:

- Certified Languages has been updated.
- The "Downloading Google Apps Third Party Libraries" section has been renamed to "Downloading and Copying Google Apps Third-Party Libraries. In addition, this section has been modified.
- Section 2.1.2.5, "Creating a Target System User Account for Connector Operations" has been modified.
- The "Enabling Administrative API for Google Apps Business Domain" section has been replaced with Section 2.1.2.6, "Enabling Administrative API for Google Apps Business Domain."
- The following sections have been added:
  - Section 2.1.2.1, "Registering the Client Application"
  - Section 2.1.2.2, "Activating API Services"
  - Section 2.1.2.3, "Creating a Service Account"
  - Section 2.1.2.4, "Adding Scopes for the Client Application"
  - Section 2.1.2.6, "Enabling Administrative API for Google Apps Business Domain"
  - Full Reconciliation
  - Guidelines on Performing Provisioning Operations

- Section 5.1.1, " Google Apps Target User Delete Recon Scheduled Job Revokes All Accounts Provisioned by the Secondary IT Resource" has been removed as it is no longer an issue.

Documentation-Specific Updates in Release 11.1.1.6.0

These are the updates made to Release 11.1.1.6.0 of the connector.

- The following documentation-specific update has been made in the revision "10" of release 11.1.1.6.0:
  - Information about limited reconciliation has been modified in Limited Reconciliation.
- The following documentation-specific update has been made in the revision "9" of release 11.1.1.6.0:
  - In Installing the Connector in Oracle Identity Manager,all instances of *11.1.1.5.0* have been replaced with *RELEASE NUMBER.*
- The following documentation-specific updates have been made in the revision "8" of release 11.1.1.6.0:
  - The "Oracle Identity Manager" row in Table 1-1 has been updated.
  - The name of the "Known Issues and Troubleshooting" chapter has been changed to "Known Issues, Workarounds, and Troubleshooting." In addition, Known Issues, Workarounds, and Troubleshooting has been restructured.
- The following documentation-specific updates have been made in the revision "7" of release 11.1.1.6.0:
  - The "Oracle Identity Manager" row in Table 1-1 has been modified.
  - A note has been added in the "Files in the dataset directory" and "xml/GoogleApps-Datasets.xml" rows of Table 2–1, " Files and Directories On the Installation Media".
  - The following sections have been added:
    * Configuring Oracle Identity Manager 11.1.2.*x* or Later
    * Localizing Field Labels in UI Forms
    * Performing Provisioning Operations in Oracle Identity Manager Release 11.1.1.*x*
    * Performing Provisioning Operations in Oracle Identity Manager Release 11.1.2.*x* or Later
  - Instructions specific to Oracle Identity Manager release 11.1.2.*x* have been added in the following sections:
    * Installing the Connector in Oracle Identity Manager
    * Configuring the IT Resource for the Target System
    * Creating the IT Resource for the Connector Server
    * Configuring Scheduled Jobs
  - Issue tracked by bug 12627688 has been added to Section 5.1, "Known Issues and Workarounds."

Documentation-Specific Updates in Release 11.1.1.5.0

These are the updates made to Release 11.1.1.5.0 of the connector.

The following documentation-specific update has been made in the revision "3" of the release 11.1.1.5.0:

- In Limited Reconciliation, the syntax of the Filter parameter has been changed.
- In Files and Directories in the Google Apps Connector Installation Package, the name of the configuration/GoogleApps-ConnectorConfig.xml file has been changed to configuration/GoogleApps-CI.xml.

The following documentation-specific update has been made in the revision "4" of the release 11.1.1.5.0:

- Installation includes connector installation scenarios depending on where you want to run the connector code (bundle), either locally in Oracle Identity Manager or remotely in a Connector Server.

The following documentation-specific update has been made in the revision "5" of the release 11.1.1.5.0:

- Creating the IT Resource for the Connector Server has been added to the connector guide to deploy the connector bundle remotely in a connector server.

The following documentation-specific update has been made in the revision "6" of the release 11.1.1.5.0:

- In Managing Logging, the logger name has been corrected.

1

About the Connector

Oracle Identity Manager automates access rights management, security, and provisioning of IT resources. Oracle Identity Manager connectors are used to integrate Oracle Identity Manager with external, identity-aware applications. This guide discusses the connector that enables you to use Google Apps as a managed (target) resource of Oracle Identity Manager.

> **Note:**
>
> At some places in this guide, the term target system has been used to refer to Google Apps.

In the account management (target resource) mode of the connector, data about users created or modified directly on the target system can be reconciled into Oracle Identity Manager. This data is used to provision (allocate) new resources or update resources already assigned to OIM Users. In addition, you can use Oracle Identity Manager to provision or update Google Apps resources (that is, accounts) assigned to OIM Users. These provisioning operations performed on Oracle Identity Manager translate into the creation of or updates to target system accounts.

This chapter contains the following sections:

1.1 Certified Components

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

Certified Components lists certified components for the connector.

Table 1-1 Certified Components

Component	Requirement
Oracle Identity Governance or Oracle Identity Manager	You can use one of the following releases of Oracle Governance or Oracle Identity Manager: • Oracle Identity Governance 12*c* (12.2.1.4.0) • Oracle Identity Governance 12*c* (12.2.1.3.0) • Oracle Identity Manager 11*g* Release 1 (11.1.1.5.0) and any later BP in this release track • Oracle Identity Manager 11*g* Release 2 (11.1.2.0.4) and any later BP in this release track • Oracle Identity Manager 11*g* Release 2 PS2 (11.1.2.2.0) and any later BP in this release track • Oracle Identity Manager 11*g* Release 2 PS3 (11.1.2.3.0)
Target systems	Google Apps
Connector Server	1.4.0
Connector Server JDK	JDK 1.6 or later

1.2 Certified Languages

The connector supports the following languages:

- 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

1.3 Connector Architecture

The connector enables you to manage accounts on the target system. Managing accounts consists of the following processes:

- Provisioning

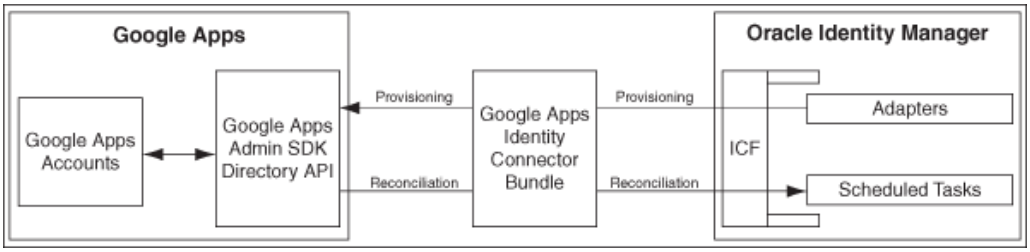
  Provisioning involves creating or updating users on the target system through Oracle Identity Manager. When you allocate (or provision) a Google Apps resource to an OIM User, the operation results in the creation of an account on Google Apps for that user. In the Oracle Identity Manager context, the term provisioning also covers updates made to the target system account through Oracle Identity Manager.

- Target resource reconciliation

  In target resource reconciliation, data related to newly created and modified target system accounts can be reconciled and linked with existing OIM Users and provisioned resources. A scheduled task is used for reconciliation.

Figure 1-1 shows Google Apps connector integration with Oracle Identity Manager.

Figure 1-1 Architecture of the Google Apps Connector



As shown in this figure, Google Apps is configured as a target resource of Oracle Identity Manager. Through provisioning operations performed on Oracle Identity Manager, accounts are created and updated on the target system for OIM Users. Through reconciliation, account data that is created and updated directly on the target system is fetched into Oracle Identity Manager and stored against the corresponding OIM Users.

Identity Connector Framework (ICF) is a component that is required in order to use Identity Connectors. ICF is distributed together with Oracle Identity Manager. You do not need to configure or modify ICF.

During provisioning, the Adapters invoke ICF operation, ICF inturn invokes create operation on Google Apps Identity Connector Bundle and then the bundle calls Google Apps Provisioning API. The Google Apps provisioning API on the target system accepts provisioning data from the bundle, carries out the required operation on the target system, and returns the response from the target system back to the bundle, which passes it to the adapters.

During reconciliation, a scheduled task invokes ICF operation, ICF inturn invokes create operation on Google Apps Identity Connector Bundle and then the bundle calls Google Apps Reconciliation API. The API extracts user records that match the reconciliation criteria and hands them over through the bundle and ICF back to the scheduled task, which brings the records to Oracle Identity Manager.

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

Each record fetched from the target system is compared with Google Apps resources that are already provisioned to OIM Users. If a match is found, then the update made to the Google Apps record from the target system is copied to the Google Apps resource in Oracle Identity Manager. If no match is found, then the user ID of the record is compared with the user ID of each OIM User. If a match is found, then data in the target system record is used to provision a Google Apps resource to the OIM User.

The Google Apps Identity Connector Bundle communicates with the Google Apps Admin SDK's Directory API using the HTTPS protocol. Internally, the library uses the `java.net.HttpURLConnection` class. When you deploy and start using the connector, it sets the following system properties for configuring the proxy for the connections created by the `HttpURLConnection` class:

- https.proxyPort
- https.proxyHost

> **Note:**
>
> Setting of these system properties might have an impact on the JVM and all other classes that use the `HttpURLConnection` class.

In addition, to support user name/password based proxy authentication, the connector provides and registers an implementation of the java.net.Authenticator class.

Depending on your application server configuration, it might be necessary to import google certificates to application server keystore/truststore.

1.4 Features of the Connector

The following are features of the connector:

- Full Reconciliation
- Batched Reconciliation
- Support for the Connector Server
- Support for Reconciliation of Account Status
- Support for Reconciliation of Deleted Account Data
- Support for Connector Operations in Multiple Domains

1.4.1 Full Reconciliation

In full reconciliation, all records are fetched from the target system to Oracle Identity Manager.

> **Note:**
>
> The connector cannot support incremental reconciliation because the target system does not provide a way for tracking the time at which account data is created or modified.

For more information, see Full Reconciliation.

1.4.2 Batched Reconciliation

You can break down a reconciliation run into batches by specifying the number of records that must be included in each batch.

For more information, see Batched Reconciliation.

1.4.3 Support for the Connector Server

Connector Server is one of the features provided by ICF. 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 want 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 if the bundle works faster when deployed on the same host as the native managed resource.

> **See Also:**
>
> Using an Identity Connector Server in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager* for information about installing and configuring connector server and running the connector server

1.4.4 Support for Reconciliation of Account Status

Support for reconciliation of account status is one of the features where the connector fetches the status information during a reconciliation operation.

During a reconciliation run, the connector can fetch status information along with the rest of the account data.

1.4.5 Support for Reconciliation of Deleted Account Data

The Google Apps Target Resource User Delete Reconciliation scheduled task can be used to fetch details of deleted target system users.

This information is used to revoke the corresponding Google Apps resources from OIM Users.

1.4.6 Support for Connector Operations in Multiple Domains

By default, this connector supports reconciliation and provisioning operations within a single domain. However, you can configure the connector for performing connector operations in more than one domain by specifying a value for the `supportMultipleDomain` entry of the Main configuration lookup definition.

For more information, see Lookup.Configuration.GoogleApps.

1.5 Lookup Definitions Used During Connector Operations

Lookup definitions used during reconciliation and provisioning are either preconfigured or can be synchronized with the target system.

Lookup definitions used during connector operations can be categorized as follows:

- Lookup Definitions Synchronized with the Target System
- Preconfigured Lookup Definitions

1.5.1 Lookup Definitions Synchronized with the Target System

Lookup field synchronization involves copying additions or changes made to specific fields in the target system to lookup definitions in Oracle Identity Manager.

During a provisioning operation, you use the Group Name lookup field on the process form to specify a group for the user for whom the provisioning operation is being performed. The Group Name lookup field is populated with values from the Lookup.GoogleApps.Groups lookup definition, which is automatically created on Oracle Identity Manager when you deploy the connector.

The Code Key and Decode columns contain values of the `__NAME__` connector attribute. The `__NAME__` attribute of both Code Key and Decode columns is mapped to Group email address in the target system.

When you perform lookup field synchronization, Groups on the target system are fetched to Oracle Identity Manager and populated in the `Lookup.GoogleApps.Groups` lookup definition. You use the Google Apps Lookup Recon scheduled task to perform

lookup field synchronization. Configuring Scheduled Jobs describes this scheduled job.

1.5.2 Preconfigured Lookup Definitions

Preconfigured lookup definitions are the other lookup definitions that are created in Oracle Identity Manager when you deploy the connector. These lookup definitions are either prepopulated with values or values must be manually entered in them after the connector is deployed.

This section discusses the other lookup definitions that are created in Oracle Identity Manager when you deploy the connector. The other lookup definitions are as follows:

- Lookup.Configuration.GoogleApps
- Lookup.GoogleApps.UM.Configuration
- Lookup.GoogleApps.UM.ProvAttrMap
- Lookup.GoogleApps.UM.ReconAttrMap
- Lookup.GoogleApps.GM.Configuration
- Lookup.GoogleApps.GM.ProvAttrMap
- Lookup.GoogleApps.GM.ReconAttrMap

1.5.2.1 Lookup.Configuration.GoogleApps

The Lookup.Configuration.GoogleApps lookup definition contains the entries shown in Table 1-2.

Table 1-2 Entries in the Lookup.Configuration.GoogleApps Lookup Definition

Code Key	Decode	Description
Bundle Name	org.identityconnectors.googleapps	This entry holds the name of the connector bundle. Do not modify this entry.
Bundle Version	1.2.1	This entry holds the version of the connector bundle. Do not modify this entry.
Connector Name	org.identityconnectors.googleapps.GoogleAppsConnector	This entry holds the name of the connector class. Do not modify this entry.
Group Configuration Lookup	Lookup.GoogleApps.GM.Configuration	This entry holds the name of the lookup definition that contains group-specific configuration properties. Do not modify this entry.
supportMultipleDomain	false	This entry specifies whether the connector is configured to perform connector operations in a single or multiple domain. By default, the connector performs connector operations only on the domain specified in IT resource. Set the value of this entry to `true` if you want the connector to perform connector operations in all the domains present in Google Apps. Otherwise, set it to `false`.
User Configuration Lookup	Lookup.GoogleApps.UM.Configuration	This entry holds the name of the lookup definition that stores configuration information used during user management operations. Do not modify this entry.

1.5.2.2 Lookup.GoogleApps.UM.Configuration

The Lookup.GoogleApps.UM.Configuration lookup definition holds configuration entries that are specific to the user object type. This lookup definition is used during user management operations.

Table 1-3 lists the default entries in this lookup definition.

Table 1-3 Entries in the Lookup.GoogleApps.UM.Configuration Lookup Definition

Code Key	Decode	Description
Provisioning Attribute Map	Lookup.GoogleApps.UM.ProvAttrMap	This entry holds the name of the lookup definition that stores attribute mappings between Oracle Identity Manager and the target system. This lookup definition is used during provisioning operations.
Recon Attribute Map	Lookup.GoogleApps.UM.ReconAttrMap	This entry holds the name of the lookup definition that stores attribute mappings between Oracle Identity Manager and the target system. This lookup definition is used during reconciliation.

1.5.2.3 Lookup.GoogleApps.UM.ProvAttrMap

The Lookup.GoogleApps.UM.ProvAttrMap lookup definitions hold mappings between process form fields (Code Key values) and target system attributes (Decode). This lookup definition is preconfigured and is used during provisioning. Table 1-4 lists the default entries.

Table 1-4 Entries in the Lookup.GoogleApps.UM.ProvAttrMap Lookup Definitions

Code Key (Process Form Field)	Decode (Google Apps Field)
Account Name	__NAME__
Change Password At Next Login	changePasswordAtNextLogin
Family Name	familyName
Given Name	givenName
Is Admin	isAdmin
OrgUnit Path	orgunitpath
Password	__PASSWORD__
UD_GA_GROUP~Group Name[Lookup]	groups
UD_GA_NICK~Nick Name	aliases
Unique Id	__UID__

1.5.2.4 Lookup.GoogleApps.UM.ReconAttrMap

The Lookup.GoogleApps.UM.ReconAttrMap lookup definition holds mappings between resource object fields (Code Key values) and target system attributes (Decode). This lookup definition preconfigured and is used during reconciliation. Table 1-5 lists the default entries.

Table 1-5 Entries in the Lookup.GoogleApps.UM.ReconAttrMap Lookup Definition

Code Key (Resource Object Field)	Decode (Google Apps Field)
Account Name	__NAME__
Change Password At Next Login	changePasswordAtNextLogin
Family Name	familyName
Given Name	givenName
Group Names~Group Name[Lookup]	groups
Is Admin	isAdmin
Nick Names~Nick Name	aliases
OrgUnit Path	orgunitpath
Status	__ENABLE__
Unique Id	__UID__

1.5.2.5 Lookup.GoogleApps.GM.Configuration

The Lookup.GoogleApps.GM.Configuration lookup definition holds configuration entries that are specific to the group object type. This lookup definition is used during group management operations when your target system is configured as a target resource.

Table 1-6 lists the default entries in this lookup definition.

Table 1-6 Entries in the Lookup.GoogleApps.GM.Configuration Lookup Definition

Code Key	Decode	Description
Provisioning Attribute Map	Lookup.GoogleApps.GM.ProvAttrMap	This entry holds the name of the lookup definition that stores attribute mappings between Oracle Identity Manager and the target system. This lookup definition is used during provisioning operations.
Recon Attribute Map	Lookup.GoogleApps.GM.ReconAttrMap	This entry holds the name of the lookup definition that stores attribute mappings between Oracle Identity Manager and the target system. This lookup definition is used during reconciliation.

1.5.2.6 Lookup.GoogleApps.GM.ProvAttrMap

The Lookup.GoogleApps.GM.ProvAttrMap lookup definition holds mappings between process form fields (Code Key values) and target system attributes (Decode). This lookup definition is preconfigured and is used during group provisioning operations. Table 1-7 lists the default entries.

Table 1-7 Entries in the Lookup.GoogleApps.GM.ProvAttrMap Lookup Definition

Group Field on Oracle Identity Manager	Google Apps Field
Allow External Members	allowExternalMembers
Description	description
Email Address	email
Group Name	name
Is Archived	isArchived
Unique Id	__UID__
Who Can Join	whoCanJoin
Who Can View Group	whoCanViewGroup
Who Can View Membership	whoCanViewMembership

1.5.2.7 Lookup.GoogleApps.GM.ReconAttrMap

The Lookup.ActiveDirectory.GM.ReconAttrMap lookup definition holds mappings between resource object fields (Code Key values) and target system attributes (Decode). This lookup definition is preconfigured and is used during target resource group reconciliation runs. Table 1-8 lists the default entries.

Table 1-8 Entries in the Lookup.GoogleApps.GM.ReconAttrMap Lookup Definition

Group Field on Oracle Identity Manager	Google Apps Field
Allow External Members	allowExternalMembers
Description	description
Email Address	email
Group Name	name
Is Archived	isArchived
OIM Org Name	Organization Name **Note:** This is a connector attribute. The value of this attribute is used internally by the connector to specify the organization of the groups in Oracle Identity Manager.
Unique Id	__UID__
Who Can Join	whoCanJoin
Who Can View Group	whoCanViewGroup
Who Can View Membership	whoCanViewMembership

1.6 Connector Objects Used During Target Resource Reconciliation

Target resource reconciliation involves fetching data about newly created or modified accounts on the target system and using this data to add or modify resources assigned to OIM Users.

The GoogleApps Target Resource User Reconciliation scheduled job is used to initiate a reconciliation run. This scheduled job is discussed in –-------

This section discusses the following topics:

- User Fields for Target Resource Reconciliation
- Group Fields for Reconciliation
- Reconciliation Rules
- Reconciliation Action Rules

1.6.1 User Fields for Target Resource Reconciliation

The Lookup.GoogleApps.UM.ReconAttrMap lookup definition maps resource object fields with target system attributes. This lookup definition is used for performing target resource user reconciliation runs.

In this lookup definition, entries are in the following format:

- **Code Key**: Reconciliation field of the resource object
- **Decode**: Name of the target system attribute

Table 1-9 lists the entries in this lookup definition.

Table 1-9 Entries in the Lookup.GoogleApps.UM.ReconAttrMap Lookup Definition

Code Key (Resource Object Field)	Decode (Google Apps Field)
Account Name	__NAME__
Change Password At Next Login	changePasswordAtNextLogin
Family Name	familyName
Given Name	givenName
Group Names~Group Name[Lookup]	groups
Is Admin	isAdmin
Nick Names~Nick Name	aliases
OrgUnit Path	orgunitpath
Status	__ENABLE__
Unique Id	__UID__

1.6.2 Group Fields for Reconciliation

The Lookup.GoogleApps.GM.ReconAttrMap lookup definition maps user resource object fields and target system attributes. This lookup definition is used for performing target resource group reconciliation runs.

Table 1-9 lists the group fields of the target system from which values are fetched during reconciliation. The GoogleApps Group Recon scheduled job is used to reconcile group data.

Table 1-10 Entries in the Lookup.GoogleApps.GM.ReconAttrMap Lookup Definition

Group Field on Oracle Identity Manager	Google Apps Field
Google Apps Field	allowExternalMembers
Description	description
Email Address	email
Group Name	name
Is Archived	isArchived
OIM Org Name	Organization Name **Note:** This is a connector attribute. The value of this attribute is used internally by the connector to specify the organization of the groups in Oracle Identity Manager.
Unique Id	__UID__
Who Can Join	whoCanJoin
Who Can View Group	whoCanViewGroup
Who Can View Membership	whoCanViewMembership

1.6.3 Reconciliation Rules

The following sections provide information about the reconciliation rules for this connector:

- Reconciliation Rule for Users
- Reconciliation Rule for Groups
- Viewing Reconciliation Rules

1.6.3.1 Reconciliation Rule for Users

The following is the process-matching rule for users:

Rule name: GoogleApps User Recon Rule

Rule element: (GAPPS User GUID Equals Unique ID) OR (User Login Equals Tokenize(Account Name))

In this first rule component:

- GAPPS User GUID is a UDF on the OIM User form that holds the unique ID of the Google Apps user.

- Unique ID is the UID field of the Google Apps account.

In this second rule component:

- User Login is the User ID field of the OIM User form.
- Tokenize(Account Name) is the name part in the email address of the Google Apps account.

1.6.3.2 Reconciliation Rule for Groups

The following is the process-matching rule for groups:

Rule name: GoogleApps Groups Recon Rule

Rule element: Organization Name Equals OIM Org Name

In this rule element:

- Organization Name is the Organization Name field of the OIM User form.
- OIM Org Name is the organization name of the groups in Oracle Identity Manager. OIM Org Name is the value specified in the Organization Name attribute of the GoogleApps Group Recon scheduled job.

1.6.3.3 Viewing Reconciliation Rules

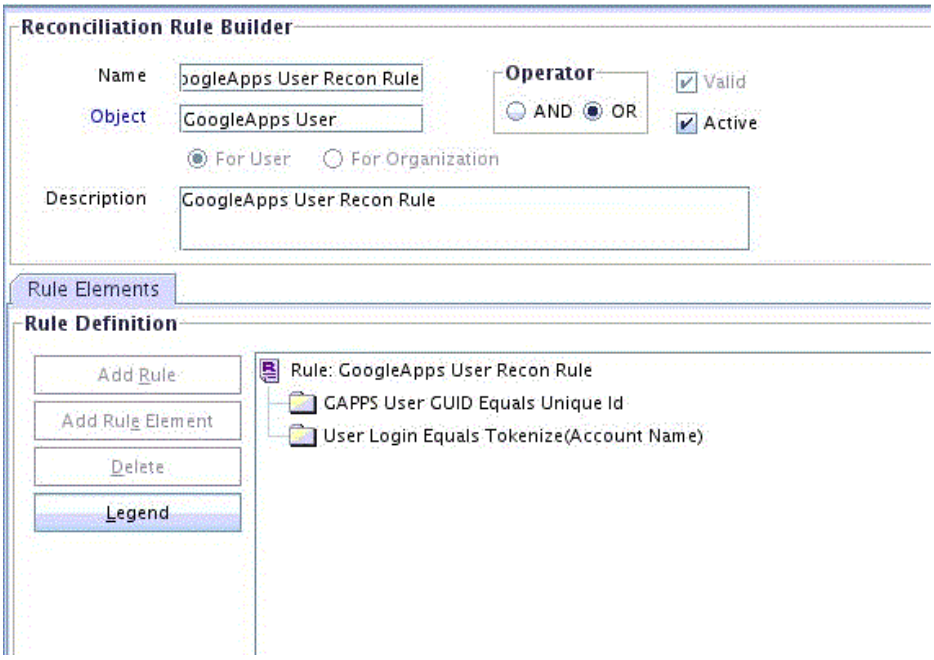
After you deploy the connector, you can view the reconciliation rule for reconciliation by performing the following steps:

> **Note:**
>
> Perform the following procedure only after the connector is deployed.

1. Log in to the Oracle Identity Manager Design Console.
2. Expand **Development Tools**.
3. Double-click **Reconciliation Rules**.
4. Search for and open one of the following:
   - For Users: **GoogleApps User Recon Rule**

     Figure 1-2 shows the reconciliation rule for users.

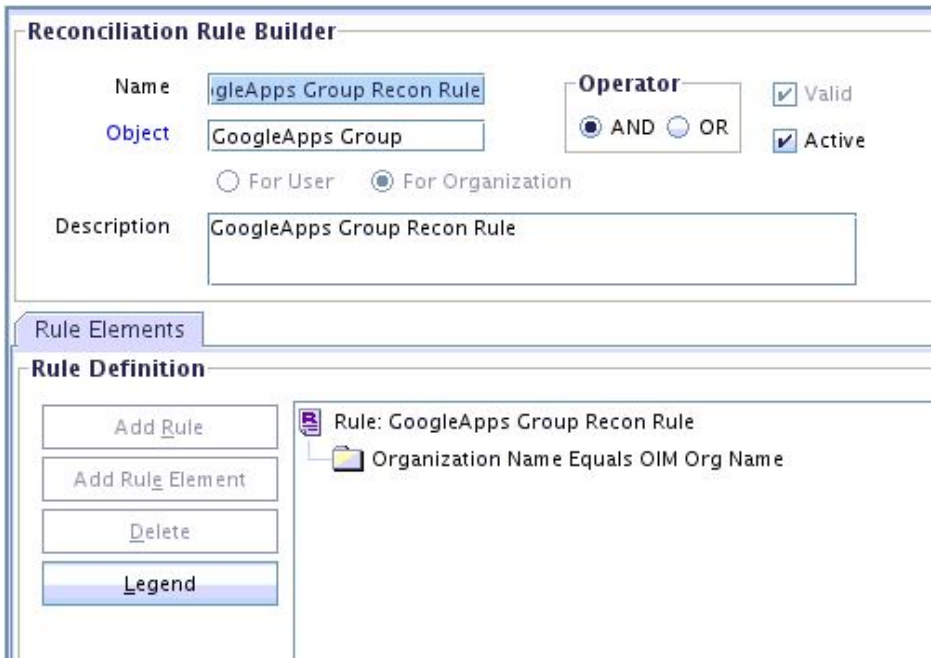
Figure 1-2 Reconciliation Rule for Users



- For Groups: **GoogleApps Groups Recon Rule**

  Figure 1-3 shows the reconciliation rule for groups.

Figure 1-3 Reconciliation Rule for Groups



1.6.4 Reconciliation Action Rules

Reconciliation action rules specify actions that must be taken depending on whether or not matching Google Apps resources or OIM Users are found when the reconciliation rule is applied.

> **Note:**
>
> No action is performed for rule conditions that are not predefined for this connector. You can define your own action rule for such rule conditions.

The following sections provide information about the reconciliation rules for this connector:

- Reconciliation Action Rules for User and Groups
- Viewing Reconciliation Action Rules

1.6.4.1 Reconciliation Action Rules for User and Groups

Table 1-11 lists the action rules for both users and groups reconciliation.

Table 1-11 Action Rules for Reconciliation

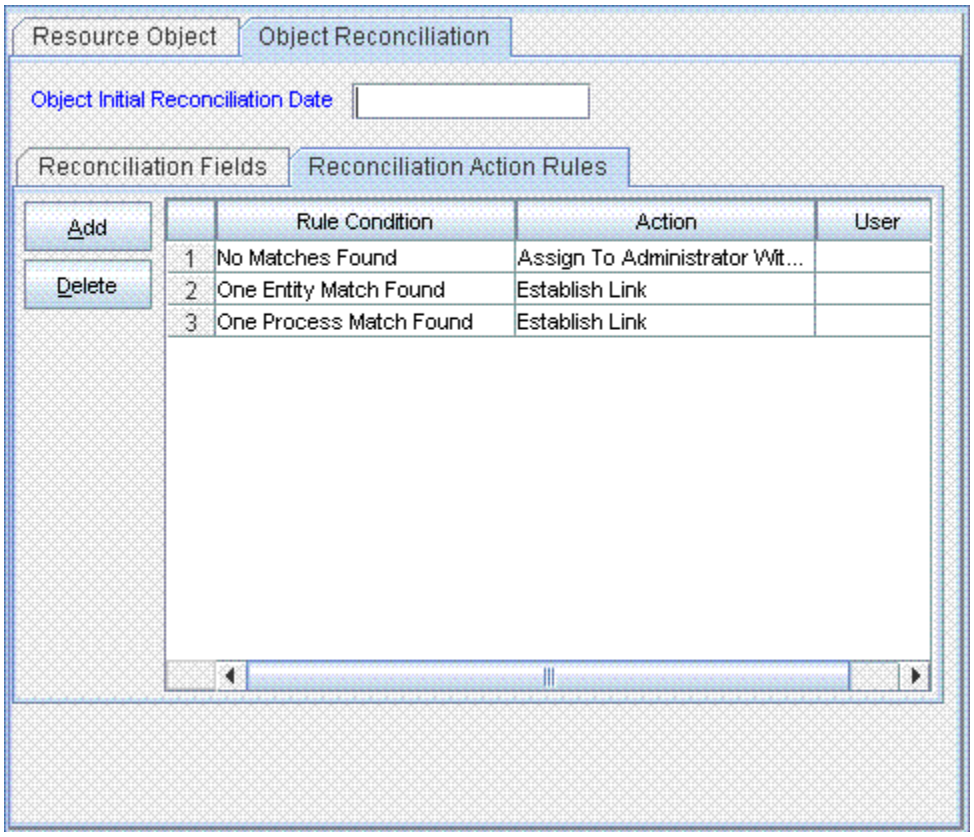
Rule Condition	Action
No Matches Found	Assign to Administrator With Least Load
One Entity Match Found	Establish Link
One Process Match Found	Establish Link

1.6.4.2 Viewing Reconciliation Action Rules

After you deploy the connector, you can view the reconciliation action rules for users and groups reconciliation by performing the following steps:

1. Log in to the Design Console.
2. Expand **Resource Management**, and double-click **Resource Objects**.
3. If you want to view the reconciliation action rules for reconciliation, then search for and open one of the following resource objects:
   - For Users: **GoogleApps User**
   - For Groups: **GoogleApps Group**
4. Click the **Object Reconciliation** tab, and then click the **Reconciliation Action Rules** tab. The Reconciliation Action Rules tab displays the action rules defined for this connector. Figure 1-4 shows the reconciliation action rules for reconciliation.

Figure 1-4 Reconciliation Action Rules



1.7 Connector Objects Used During Provisioning

Provisioning involves creating or modifying user data on the target system through Oracle Identity Manager.

This section discusses the following topics:

- Provisioning Functions
- User Fields for Provisioning
- Group Fields for Provisioning

1.7.1 Provisioning Functions

These are the supported provisioning functions and the adapters that perform these functions for the Google Apps connector.

Table 1-12 lists the supported provisioning functions and the adapters that perform these functions. The Adapter column gives the name of the adapter that is used when the function is performed.

> **See Also:**
>
> Process Task Adapters in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager* for generic information about process tasks and adapters

Table 1-12 User Provisioning Functions

Function	Adapter
Create User	adpGOOGLEAPPSCREATEOBJECT
Update User	adpGOOGLEAPPSUPDATEATTRIBUTEVALUE
Delete user	adpGOOGLEAPPSDELETEOBJECT
Enable user	adpGOOGLEAPPSENABLEUSER
Disable user	adpGOOGLEAPPSDISABLEUSER
Change or reset password	adpGOOGLEAPPSUPDATEATTRIBUTEVALUE
Update child table values	adpGOOGLEAPPSUPDATECHILDTABLEVALUE
Add child table values	adpGOOGLEAPPSADDCHILDTABLEVALUES
Remove child table values for a user	adpGOOGLEAPPSREMOVECHILDTABLEVALUES

1.7.2 User Fields for Provisioning

The Lookup.GoogleApps.UM.ProvAttrMap lookup definition maps process form fields with Google Apps fields. This lookup definition is used for performing user provisioning operations.

In this lookup definition, entries are in the following format:

- **Code Key**: Name of the process form field
- **Decode**: Name of the target system attribute

Table 1-13 lists the entries in this lookup definition.

Table 1-13 Entries in the Lookup.GoogleApps.UM.ProvAttrMap Lookup Definitions

Code Key (Process Form Field)	Decode (Google Apps Field)
Account Name	__NAME__
Change Password At Next Login	changePasswordAtNextLogin
Family Name	familyName
Given Name	givenName
Is Admin	isAdmin
OrgUnit Path	orgunitpath
Password	__PASSWORD__
UD_GA_GROUP~Group Name[Lookup]	groups
UD_GA_NICK~Nick Name	aliases
Unique Id	__UID__

1.7.3 Group Fields for Provisioning

The Lookup.GoogleApps.GM.ProvAttrMap lookup definition maps user resource object fields and target system attributes. This lookup definition is used for performing group provisioning operations.

Table 1-14 lists the group fields of the target system for which you can specify or modify values during provisioning operations.

Table 1-14 Entries in the Lookup.GoogleApps.GM.ProvAttrMap Lookup Definition

Group Field on Oracle Identity Manager	Google Apps Field
Allow External Members	allowExternalMembers
Description	description
Email Address	email
Group Name	name
Is Archived	isArchived
Unique Id	__UID__
Who Can Join	whoCanJoin
Who Can View Group	whoCanViewGroup
Who Can View Membership	whoCanViewMembership

2
Deploying the Google Apps Connector

The procedure to deploy the connector is divided across three stages namely preinstallation, installation, and postinstallation.

The following topics provide details on these stages:

- Preinstallation
- Installation
- Postinstallation
- Upgrading the Connector

> **Note:**
>
> Some of the procedures described in this chapter must be performed on the target system. To perform these procedures, you must use a Google Apps account with administrator privileges.

2.1 Preinstallation

Preinstallation involves copying third-party libraries to the computer hosting Oracle Identity Manager. It also involves registering the connector with Google Apps for accessing user management APIs and creating a target system account for the connector.

The following topics provide details on these preinstallation procedures:

- Downloading and Copying Google Apps Third-Party Libraries
- Preinstallation on the Target System

2.1.1 Downloading and Copying Google Apps Third-Party Libraries

Perform the following steps to download and copy Google Apps third-party libraries:

1. Download Google Apps third-party libraries as following:
   a. Visit the following URL:

      https://developers.google.com/

   b. In the API Guides, Reference, and Client Libs region, click **client libraries**.
   c. On the Directory API: Client Libraries page, in the "Client library" column of the table, click **Google APIs Client Library for Java**.
   d. Save the ZIP file to a temporary location. This file contains the following third-party libraries:

i. google-api-client-1.18.0-rc.jar

ii. google-api-services-admin-directory_v1-VERSION.jar

iii. In this file name, VERSION is the latest version of the JAR file available in the ZIP file. For example, `google-api-services-admin-directory_v1-rev35-1.18.0-rc.jar`.

iv. google-api-services-groupssettings-v1-[version].jar

In this file name, VERSION is the latest version of this JAR file that is available in the ZIP file. For example, `google-api-services-groupssettings-v1-rev43-1.17.0-rc.jar`.

If this file is not available, then visit the following URL and click **Download the Groups Settings API v1 Client Library for Java** to obtain the ZIP file that contains this JAR.

https://developers.google.com/api-client-library/java/apis/groupssettings/v1

v. google-http-client-1.18.0-rc.jar

vi. google-http-client-jackson2-1.18.0-rc.jar

vii. google-oauth-client-1.18.0-rc.jar

viii. httpclient-4.0.1.jar

ix. httpcore-4.0.1.jar

x. jackson-core-2.1.3.jar

> **Note:**
>
> If the specified versions of the JAR files listed in this step are not available, then you can obtain them by visiting the following URL and then searching for and downloading the JAR files in the Downloads tab:
>
> https://code.google.com/p/google-api-java-client

2. Copy the downloaded third-party libraries as follows:

a. Create a directory named googleapps-***RELEASE_NUMBER*** under the following directory.

OIM_HOME/server/ConnectorDefaultDirectory/targetsystems-lib/

For example, if you are using release 11.1.1.7.0 of this connector, then create a directory named googleapps-11.1.1.7.0 in the *OIM_HOME*/server/ConnectorDefaultDirectory/targetsystems-lib/ directory.

b. Copy the third-party libraries downloaded in Step 1 to the *OIM_HOME*/server/ConnectorDefaultDirectory/targetsystems-lib/googleapp-***RELEASE_NUMBER*** directory.

2.1.2 Preinstallation on the Target System

This section provides a high-level summary about the preinstallation tasks to be performed on the target system.

The preinstallation process involves performing the following tasks:

> **Note:**
>
> The detailed instructions for performing each of these preinstallation tasks are available in the Google Cloud Platform Documentation at `https://cloud.google.com/docs/`

1. Create a project and register your client application with the Google Apps Cloud platform in the Google Developers Console.
2. Activate the associated API services such as adding custom information, enable billing, and page monitoring services, for your client application. While activating the associated API services ensure that the statuses of the **Admin SDK** and **Group Settings APIs** are set to ON.
3. Create a service account and enable your client application to access the activated APIs. Additionally, create a Client ID, Public/Private key pair, and password for the earlier created service account. After the service account creation, note down the Client ID, Public/Private key pair and password information. This information is required while adding scopes and also while configuring the IT resource parameters.
4. Add scopes and authorize the registered client application.
5. Create a user account on the target system. The connector uses this account to connect to the target system during each connector operation. Post account creation, assign the **Groups Admin** and **User Management Admin** admin roles to the newly created account.
6. Enable access to various Google administrative APIs available in the Google Apps Business Domain. The administrative API allows you to manage user accounts and synchronizes Google Apps user accounts with your own user account
7. Enable external user access to groups in Google Apps. Perform this step only if you want external users to access groups in Google Apps.

2.2 Installation

You must install the connector in Oracle Identity Manager. If necessary, you can also deploy the connector in a Connector Server.

Installation information is divided across the following sections:

- Understanding Installation of the Google Apps Connector
- Installing the Connector in Oracle Identity Manager

2.2.1 Understanding Installation of the Google Apps Connector

Depending on where you want to run the connector code (bundle), the connector provides installation options:

The following are the installation options:

- Run the connector code locally in Oracle Identity Manager.

  To run the connector code locally in Oracle Identity Manager, perform the procedure described in Installing the Connector in Oracle Identity Manager.

- Run the connector code remotely in a Connector Server.

  To run the connector code remotely in a Connector Server, perform the procedures described in Installing the Connector in Oracle Identity Manager and Deploying the Connector Bundle in a Connector Server.

2.2.2 Installing the Connector in Oracle Identity Manager

Perform this procedure to install the Google Apps connector in Oracle Identity Manager.

> **Note:**
>
> In this guide, the term **Connector Installer** has been used to refer to the Install Connectors feature of the Administrative and User Console.

To run the Connector Installer:

1. Copy the contents of the connector installation media into the following directory:

   OIM_HOME/server/ConnectorDefaultDirectory

2. If you have not already done so, create a directory in OIM_HOME/ConnectorDefaultDirectory/targetsystems-lib with the same name as the installer package. For example:

   OIM_HOME/server/ConnectorDefaultDirectory/targetsystems-lib/googleapps-11.1.1.7.0

   Copy the third-party libraries to this directory. See Downloading and Copying Google Apps Third-Party Libraries.

3. If you are using Oracle Identity Manager release 11.1.1.*x:*

   a. Log in to Oracle Identity Manager Administrative and User Console by using the user account described in Creating the User Account for Installing Connectors of *Oracle Fusion Middleware Administering Oracle Identity Manager*.

   b. On the Welcome to Identity Manager Advanced Administration page, in the System Management region, click **Manage Connector.**

4. If you are using Oracle Identity Manager release 11.1.2.*x* or later:

a. Log in to Oracle Identity System Administration by using the user account described in Creating the User Account for Installing Connectors of *Oracle Fusion Middleware Administering Oracle Identity Manager*.

b. In the left pane, under System Management, click **Manage Connector.**

5. In the Manage Connector page, click **Install.**

6. From the Connector List list, select **GoogleApps Connector *RELEASE NUMBER***. This list displays the names and release numbers of connectors whose installation files you copy into the default connector installation directory:

   OIM_HOME/server/ConnectorDefaultDirectory

   If you have copied the installation files into a different directory, then:

   a. In the **Alternative Directory** field, enter the full path and name of that directory.

   b. To repopulate the list of connectors in the Connector List list, click **Refresh.**

   c. From the Connector List list, select **GoogleApps Connector *RELEASE NUMBER.***

7. Click **Load.**

8. To start the installation process, click **Continue.**

   The following tasks are performed in sequence:

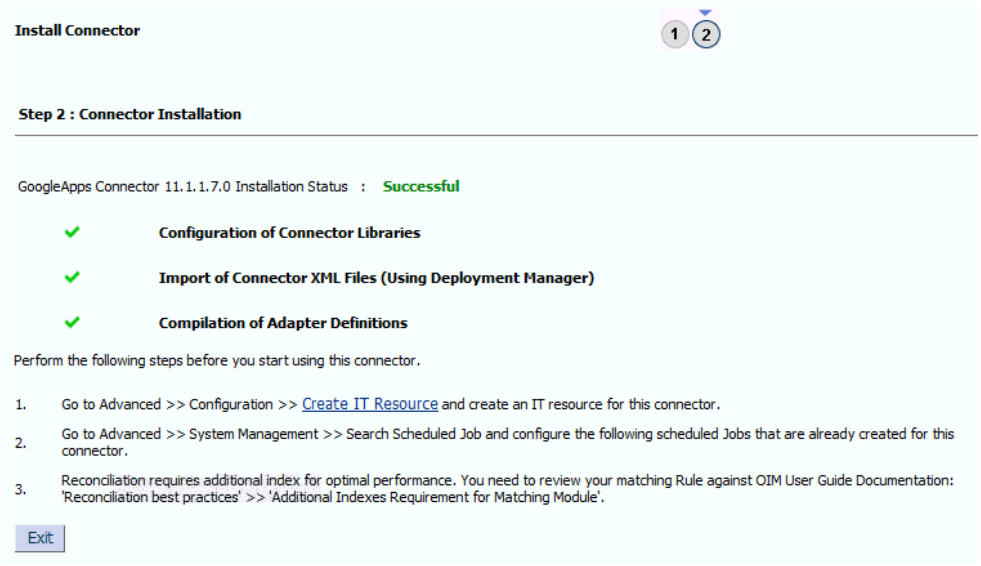
   a. Configuration of connector libraries

   b. Import of the connector XML files (by using the Deployment Manager)

   c. Compilation of adapters

   On successful completion of a task, a check mark is displayed for the task. If a task fails, then an X mark and a message stating the reason for failure are displayed. If a task fails, then make the required correction and perform one of the following steps:

Figure 2-1 Installation Status



- Retry the installation by clicking **Retry.**
- Cancel the installation and begin again from Step 3.

9. If all three tasks of the connector installation process are successful, then a message indicating successful installation is displayed.

   In addition, a list of the steps that you must perform after the installation is displayed. These steps are as follows:

   a. Ensuring that the prerequisites for using the connector are addressed

   > **Note:**
   >
   > At this stage, run the PurgeCache utility to load the server cache with content from the connector resource bundle in order to view the list of prerequisites. See Clearing Content Related to Connector Resource Bundles from the Server Cache for information about running the PurgeCache utility.
   >
   > There are no prerequisites for some predefined connectors.

   b. Configuring the IT resource for the connector.

      Record the name of the IT resource displayed on this page. The procedure to configure the IT resource is described later in this guide.

   c. Configuring the scheduled tasks that are created when you installed the connector

      Record the names of the scheduled tasks displayed on this page. The procedure to configure these scheduled tasks is described later in this guide.

   When you run the Connector Installer, it copies the connector files and external code files to destination directories on the Oracle Identity Manager host computer. These files are listed in Table A-1

2.2.3 Deploying the Connector Bundle in a Connector Server

You can deploy the Google Apps connector bundle into the Java Connector Server by performing the procedure mentioned here.

> **See Also:**
>
> Using an Identity Connector Server in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager* for information about installing and configuring connector server and running the connector server

If you want to deploy the Google Apps connector bundle into the Java Connector Server, then follow these steps:

1. Stop the Java Connector Server.

> **Note:**
>
> You can download the necessary Java Connector Server from the Oracle Technology Network web page.

2. Copy the Google Apps connector bundle into the Java Connector Server *CONNECTOR_SERVER_HOME*\bundles directory.
3. Copy Google Apps third-party libraries to the *CONNECTOR_SERVER_HOME*\lib directory. See Downloading and Copying Google Apps Third-Party Libraries for more information about the third-party libraries.
4. Start the Java Connector Server.

2.3 Postinstallation

Postinstallation for the connector involves configuring Oracle Identity Manager, enabling logging to track information about all connector events, and configuring SSL. It also involves performing some optional configurations such as enabling request-based provisioning and localizing the user interface.

Postinstallation steps are divided across the following sections:

- Configuring Oracle Identity Manager 11.1.2.*x* or Later
- Enabling Request-Based Provisioning
- Changing to the Required Input Locale
- Clearing Content Related to Connector Resource Bundles from the Server Cache
- Managing Logging
- Configuring the IT Resource for the Target System
- Creating the IT Resource for the Connector Server
- Localizing Field Labels in UI Forms

2.3.1 Configuring Oracle Identity Manager 11.1.2.*x* or Later

If you are using Oracle Identity Manager release 11.1.2.*x* or later, you must create additional metadata such as a UI form and an application instance. In addition, you must run entitlement and catalog synchronization jobs.

These procedures are described in the following sections:

- Creating and Activating a Sandbox
- Creating a New UI Form
- Creating an Application Instance
- Publishing a Sandbox
- Harvesting Entitlements and Sync Catalog
- Updating an Existing Application Instance with a New Form .

2.3.1.1 Creating and Activating a Sandbox

Create and activate a sandbox as follows. For detailed instructions, see Managing Sandboxes in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager*.

1. On the upper navigation bar, click **Sandboxes.** The Manage Sandboxes page is displayed.
2. On the toolbar, click **Create Sandbox.** The Create Sandbox dialog box is displayed.
3. In the Sandbox Name field, enter a name for the sandbox. This is a mandatory field.
4. In the Sandbox Description field, enter a description of the sandbox. This is an optional field.
5. Click **Save and Close.** A message is displayed with the sandbox name and creation label.
6. Click **OK.** The sandbox is displayed in the Available Sandboxes section of the Manage Sandboxes page.
7. Select the sandbox that you created.
8. From the table showing the available sandboxes in the Manage Sandboxes page, select the newly created sandbox that you want to activate.
9. On the toolbar, click **Activate Sandbox.**

   The sandbox is activated.

2.3.1.2 Creating a New UI Form

Create a new UI form as follows. For detailed instructions, see Managing Forms in *Oracle Fusion Middleware Administering Oracle Identity Manager*.

1. In the left pane, under Configuration, click **Form Designer.**
2. Under Search Results, click **Create.**
3. Select the resource type for which you want to create the form.
4. Enter a form name and click **Create.**

2.3.1.3 Creating an Application Instance

Create an application instance as follows. For detailed instructions, see Managing Application Instances in *Oracle Fusion Middleware Administering Oracle Identity Manager*.

1. In the System Administration page, under Configuration in the left pane, click **Application Instances.**
2. Under Search Results, click **Create.**
3. Enter appropriate values for the fields displayed on the Attributes form and click **Save.**
4. In the Form drop-down list, select the newly created form and click **Apply.**

5. Publish the application instance for a particular organization.

> **Note:**
>
> If you are using access policy-based provisioning, then specify the Active Directory connector application instance as the value for the Parent AppInstance attribute.

2.3.1.4 Publishing a Sandbox

To publish the sandbox that you created in Creating and Activating a Sandbox:

1. Close all the open tabs and pages.
2. In the upper right corner of the page, click the **Sandboxes** link. The Manage Sandboxes page is displayed.
3. From the table showing the available sandboxes in the Manage Sandboxes page, select the sandbox that you created in Creating and Activating a Sandbox.
4. On the toolbar, click **Publish Sandbox**. A message is displayed asking for confirmation.
5. Click **Yes** to confirm. The sandbox is published and the customizations it contained are merged with the main line.

2.3.1.5 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. Run the scheduled jobs for lookup field synchronization listed in –------.
2. Run the Entitlement List scheduled job to populate Entitlement Assignment schema from child process form table. See Predefined Scheduled Tasks in *Oracle Fusion Middleware Administering Oracle Identity Manager* for more information about this scheduled job.
3. Run the Catalog Synchronization Job scheduled job. See for more information about this scheduled job. See Predefined Scheduled Tasks in *Oracle Fusion Middleware Administering Oracle Identity Manager* for more information about this scheduled job.

2.3.1.6 Updating an Existing Application Instance with a New Form

For any changes you do in the Form Designer, 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 a sandbox and activate it as described in Creating and Activating a Sandbox.
2. Create a new UI form for the resource as described in Creating a New UI Form.

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 as described in Publishing a Sandbox.

2.3.2 Enabling Request-Based Provisioning

In request-based provisioning, an end user creates a request for a resource or entitlement by using the Administrative and User Console.

> **Note:**
>
> Perform the procedure described in this section *only* if you are using Oracle Identity Manager release 11.1.1.*x*.

In request-based provisioning, an end user creates a request for a resource or entitlement by using the Administrative and User Console. Administrators or other users cannot create requests for a particular user. Requests can be viewed and approved by approvers designated in Oracle Identity Manager.

> **Note:**
>
> The direct provisioning feature of the connector is automatically disabled when you enable request-based provisioning. Therefore, do not enable request-based provisioning if you want to use the direct provisioning.

To enable request-based provisioning, perform the following procedures:

- Importing Request Datasets
- Enabling the Auto Save Form Feature
- Running the PurgeCache Utility

2.3.2.1 Importing Request Datasets

There are two ways of importing request datasets:

- Importing Request Datasets Using MDS Import Utility
- Importing Request Datasets Using Deployment Manager

> **Note:**
>
> Request Datasets imported either into MDS or by using Deployment Manager are same.

2.3.2.1.1 Importing Request Datasets Using MDS Import Utility

To import a request dataset definition into the MDS:

1. Copy the predefined request datasets from the installation media to any directory on the Oracle Identity Manager host computer. The following is the list of predefined request datasets available in the dataset directory on the installation media:

   - ProvisionResource_GoogleAppsUser.xml
   - ModifyProvisionedResource_GoogleAppsUser.xml

   It is recommended that you create a directory structure as follows:

   /custom/connector/*RESOURCE_NAME*

   For example:

   E:\MyDatasets\custom\connector\GoogleApps

   > **Note:**
   >
   > Until you complete the procedure to configure request-based provisioning, ensure that there are no other files or directories inside the parent directory in which you create the directory structure. In the preceding example, ensure that there are no other files or directories inside the E:\MyDatasets directory.

   The directory structure to which you copy the predefined request dataset files is the MDS location into which this file is imported after you run the Oracle Identity Manager MDS Import utility.

2. Ensure that you have set the environment for running the MDS Import utility. See Setting up the Environment for MDS Utilities in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager* for detailed information about setting up the environment for MDS utilities.

   > **Note:**
   >
   > While setting up the properties in the weblogic.properties file, ensure that the value of the metadata_from_loc property is the parent directory of the /custom/connector/*RESOURCE_NAME* directory. For example, while performing Step 1 of this procedure, if you copy the files to the E:\MyDatasets\custom\connector\GoogleApps directory, then set the value of the metada_from_loc property to `E:\MyDatasets`.

3. In a command window, change to the *OIM_HOME*/server/bin directory.
4. Run one of the following commands:

   - On Microsoft Windows

     ```
     weblogicImportMetadata.bat
     ```

- On UNIX

  `weblogicImportMetadata.sh`

5. When prompted, enter values for the following:
   - `Please enter your username [weblogic]`

     Enter the username used to log in to the Oracle WebLogic Server

     Sample value: `WL_User`
   - `Please enter your password [weblogic]`

     Enter the password used to log in to the Oracle WebLogic Server
   - `Please enter your server URL [t3://localhost:7001]`

     Enter the URL of the application server in the following format:

     `t3://`*`HOST_NAME_IP_ADDRESS`*`:`*`PORT`*

     In this format, replace:

     HOST_NAME_IP_ADDRESS with the host name or IP address of the computer on which Oracle Identity Manager is installed.

     PORT with the port on which Oracle Identity Manager is listening.

   The request dataset is imported into the MDS.

2.3.2.1.2 Importing Request Datasets Using Deployment Manager

The request datasets (predefined or generated) can also be imported using Deployment Manager (DM), which are stored in xml/GoogleApps-Datasets.xml.

To import a request dataset definition using Deployment Manager:

1. Log in to Oracle Identity Manager Administrative and User Console.
2. Go to Advanced Administration.
3. Click **Import Deployment Manager File**.
4. Navigate to GoogleApps-Datasets.xml file and click **Add** to add it for import. The datasets available for import will be displayed.
5. Click **Import**. A message with the successful import of the datasets is displayed.

2.3.2.2 Enabling the Auto Save Form Feature

To enable the Auto Save Form feature:

1. Log in to the Design Console.
2. Expand **Process Management,** and then double-click **Process Definition.**
3. Search for and open the **GoogleApps User** process definition.
4. Select the **Auto Save Form** check box.
5. Click **Save**.

2.3.2.3 Running the PurgeCache Utility

Run the PurgeCache utility to clear content belonging to the Metadata category from the server cache. See Clearing Content Related to Connector Resource Bundles from the Server Cache for instructions.

The procedure to enable request-based provisioning ends with this step.

2.3.3 Changing to the Required Input Locale

Changing to the required input locale (language and country setting) involves installing the required fonts and setting the required input locale.

You may require the assistance of the system administrator to change to the required input locale.

2.3.4 Clearing Content Related to Connector Resource Bundles from the Server Cache

When you deploy the connector, the resource bundles are copied from the resources directory on the installation media into the Oracle Identity Manager database.

Whenever you add a new resource bundle to the connectorResources directory or make a change in an existing resource bundle, you must clear content related to connector resource bundles from the server cache.

To clear content related to connector resource bundles from the server cache:

1. In a command window, switch to the *OIM_HOME*/server/bin directory.
2. Enter one of the following commands:

> **Note:**
>
> - You can use the PurgeCache utility to purge the cache for any content category. Run `PurgeCache.bat` ***CATEGORY_NAME*** on Microsoft Windows or `PurgeCache.sh` ***CATEGORY_NAME*** on UNIX. The ***CATEGORY_NAME*** argument represents the name of the content category that must be purged.
>
> For example, the following commands purge Metadata entries from the server cache:
>
> `PurgeCache.bat MetaData`
>
> `PurgeCache.sh MetaData`
>
> - Before running the PurgeCache utility, ensure the WL_HOME and JAVA_HOME environment variables are set.

On Microsoft Windows: `PurgeCache.bat All`

On UNIX: `PurgeCache.sh All`

When prompted, enter the user name and password of an account belonging to the SYSTEM ADMINISTRATORS group. In addition, you are prompted to enter the service URL in the following format:

```
t3://OIM_HOST_NAME:OIM_PORT_NUMBER
```

In this format:

- Replace *OIM_HOST_NAME* with the host name or IP address of the Oracle Identity Manager host computer.
- Replace *OIM_PORT_NUMBER* with the port on which Oracle Identity Manager is listening.

2.3.5 Managing Logging

Oracle Identity Governance uses Oracle Java Diagnostic Logging (OJDL) for recording all types of events pertaining to the connector. OJDL is based on java.util.logger.

The following topics provide detailed information about logging:

- Understanding Log Levels
- Enabling Logging

2.3.5.1 Understanding Log Levels

> **Note:**
>
> In an Oracle Identity Manager cluster, perform this procedure on each node of the cluster. Then, restart each node.

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

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 Manager 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 2-1.

Table 2-1 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
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_SEVER* are the domain name and server name specified during the installation of Oracle Identity Manager.

2.3.5.2 Enabling Logging

Perform this procedure to enable logging in Oracle WebLogic Server.

1. Edit the logging.xml file as follows:
   a. Add the following blocks in the file:

```
<log_handler name='googleapps-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.GOOGLEAPPS" level="[LOG_LEVEL]"
useParentHandlers="false">
     <handler name="googleapps-handler"/>
     <handler name="console-handler"/>
   </logger>
```

b. Replace both occurrences of **[LOG_LEVEL]** with the ODL message type and level combination that you require. Table 2-1 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='googleapps-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="ORG.IDENTITYCONNECTORS.GOOGLEAPPS" level="NOTIFICATION:1"
useParentHandlers="false">
     <handler name="googleapps-handler"/>
     <handler name="console-handler"/>
   </logger>
```

   With these sample values, when you use Oracle Identity Manager, 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. Set the following environment variable to redirect the server logs to a file:
   - For Microsoft Windows:

     ```
     set WLS_REDIRECT_LOG=FILENAME
     ```

   - For UNIX:

     ```
     export WLS_REDIRECT_LOG=FILENAME
     ```

   Replace ***FILENAME*** with the location and name of the file to which you want to redirect the output.
4. Restart the application server.

2.3.6 Configuring the IT Resource for the Target System

The GoogleApps IT resource is automatically created when you run the Connector Installer. You must specify values for the parameters of the IT resource.

The following section describes the parameters of the IT resource:

To specify values for the parameters of the IT resource:

1. If you are using Oracle Identity Manager release 11.1.1.*x*, then:
   a. Log in to the Administrative and User Console.

b. On the Welcome page, click **Advanced** in the upper-right corner of the page.

c. On the Welcome to Oracle Identity Manager Advanced Administration page, in the Configuration region, click **Manage IT Resource**.

2. If you are using Oracle Identity Manager release 11.1.2.*x* or later, then:

a. Log in to Identity System Administration.

b. Create and activate a sandbox. For detailed instructions on creating and activating a sandbox, see Managing Sandboxes in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager*.

c. In the left pane, under Configuration, click **IT Resource.**

3. In the IT Resource Name field on the Manage IT Resource page, enter **GoogleApps** and then click **Search**.

4. Click **Edit** for the IT resource.

5. From the list at the top of the page, select **Details and Parameters**.

6. Specify values for the parameters of the IT resource. Table 2-2 describes each parameter.

> **Note:**
>
> Entries in this table are sorted in alphabetical order of parameter names.

Table 2-2 Parameters of the IT Resource

Parameter	Description
applicationName	Name of the project that was created as part of registering the client application.
Configuration Lookup	Name of the lookup definition that stores configuration information used during reconciliation and provisioning. Default value: `Lookup.Configuration.GoogleApps`
Connector Server Name	If you are using Google Apps Connector together with Java Connector Server, then provide the name of Connector Server IT Resource here.
domainName	Domain name of your Google Apps domain. Sample value: `mydomain.com`
proxyHost	The proxy host name. This is useful when a connector is to be used in the network protected by the web proxy. You can check with your network administrator for more information about proxy configuration.
proxyPassword	The proxy password. This is useful when a connector is to be used in the network protected by the web proxy. You can check with your network administrator for more information about proxy configuration.
proxyPort	The proxy port number. This is useful when a connector is to be used in the network protected by the web proxy. You can check with your network administrator for more information about proxy configuration.
proxyUsername	The proxy user name. This is useful when a connector is to be used in the network protected by the web proxy. You can check with your network administrator for more information about proxy configuration.

Table 2-2 (Cont.) Parameters of the IT Resource

Parameter	Description
scopes	The scope of your client application. Default value: `"https://www.googleapis.com/auth/admin.directory.user","https://www.googleapis.com/auth/admin.directory.group","https://www.googleapis.com/auth/admin.directory.group.member","https://www.googleapis.com/auth/apps.groups.settings"`
serviceAccountId	The email address of the service account created.
serviceAccountPrivateKey	Name and complete path to the directory containing the private key. This is the same location to which the private key was saved earlier as described in Preinstallation on the Target System. Sample value: `/scratch/34567890sdfghjk.p12`
serviceAccountUser	Name of the account used to log in to the client application. Enter the user name of account that you created. Sample value: `admin@mydomain.com`

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

2.3.7 Creating the IT Resource for the Connector Server

Perform the procedure described in this section only if you have deployed the connector bundle remotely in a Connector Server.

> **Note:**
>
> Before you deploy the connector bundle remotely in a Connector Server, you must deploy the connector in Oracle Identity Manager by performing the procedures described in Installation .

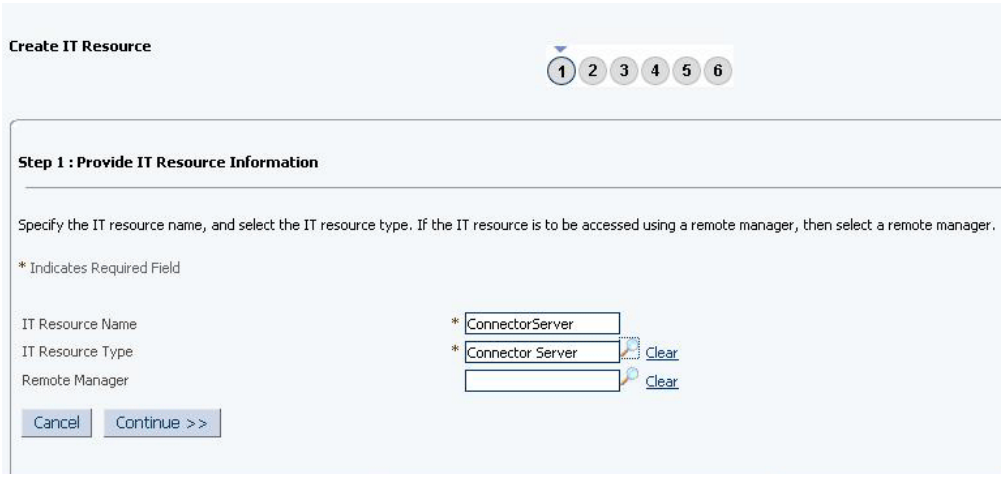
To create the IT resource for the Connector Server:

1. Depending on the Oracle Identity Manager release you are using, perform one of the following steps:
   - For Oracle Identity Manager release 11.1.1.*x*: Log in to the Administrative and User Console.
   - For Oracle Identity Manager release 11.1.2.*x* or later: Log in to Identity System Administration.
2. If you are using Oracle Identity Manager release 11.1.1.*x*, then:
   a. On the Welcome page, click **Advanced** in the upper-right corner of the page.
   b. On the Welcome to Oracle Identity Manager Advanced Administration page, in the Configuration region, click **Manage IT Resource**.
3. If you are using Oracle Identity Manager release 11.1.2.*x* or later, then in the left pane, under Configuration, click IT Resource.
4. On the Step 1: Provide IT Resource Information page, perform the following steps:
   - **IT Resource Name**: Enter a name for the IT resource.

- **IT Resource Type**: Select **Connector Server** from the IT Resource Type list.
- **Remote Manager**: Do not enter a value in this field.

5. Click **Continue**. Figure 2-2 shows the IT resource values added on the Create IT Resource page.

Figure 2-2 Step 1: Provide IT Resource Information



6. On the Step 2: Specify IT Resource Parameter Values page, specify values for the parameters of the IT resource and then click **Continue**. Figure 2-3 shows the Step 2: Specify IT Resource Parameter Values page.

Figure 2-3 Step 2: Specify IT Resource Parameter Values

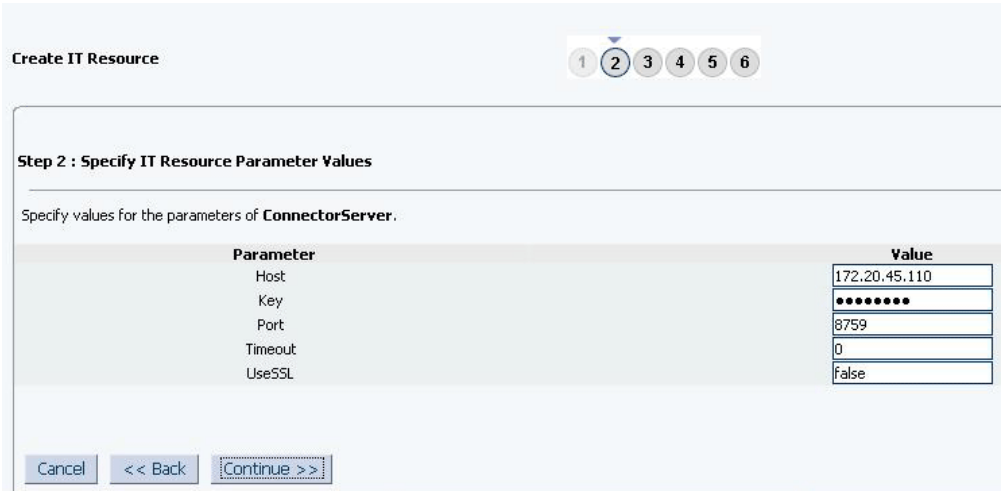


Table 2-3 provides information about the parameters of the IT resource.

Table 2-3 Parameters of the IT Resource for the Connector Server

Parameter	Description
Host	Enter the host name or IP address of the computer hosting the connector server. Sample value: `RManager`
Key	Enter the key for the Java connector server.
Port	Enter the number of the port at which the connector server is listening. Default value: `8759`
Timeout	Enter an integer value which specifies the number of milliseconds after which the connection between the connector server and Oracle Identity Manager times out. Sample value: `300`
UseSSL	Enter `true` to specify that you will configure SSL between Oracle Identity Manager and the Connector Server. Otherwise, enter `false`. Default value: `false` **Note:** It is recommended that you configure SSL to secure communication with the connector server. To configure SSL, see Configuring SSL for Java Connector Server in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*.

7. On the Step 3: Set Access Permission to IT Resource page, the `SYSTEM ADMINISTRATORS` group is displayed by default in the list of groups that have Read, Write, and Delete permissions on the IT resource that you are creating.

> **Note:**
>
> This step is optional.

If you want to assign groups to the IT resource and set access permissions for the groups, then:

a. Click **Assign Group.**

b. For the groups that you want to assign to the IT resource, select **Assign** and the access permissions that you want to set. For example, if you want to assign the ALL USERS group and set the Read and Write permissions to this group, then you must select the respective check boxes in the row, as well as the Assign check box, for this group.

c. Click **Assign.**

8. On the Step 3: Set Access Permission to IT Resource page, if you want to modify the access permissions of groups assigned to the IT resource, then:

> **Note:**
>
> - This step is optional.
> - You cannot modify the access permissions of the `SYSTEM ADMINISTRATORS` group. You can modify the access permissions of only other groups that you assign to the IT resource.

a. Click **Update Permissions.**

b. Depending on whether you want to set or remove specific access permissions for groups displayed on this page, select or deselect the corresponding check boxes.

c. Click **Update**.

9. On the Step 3: Set Access Permission to IT Resource page, if you want to unassign a group from the IT resource, then:

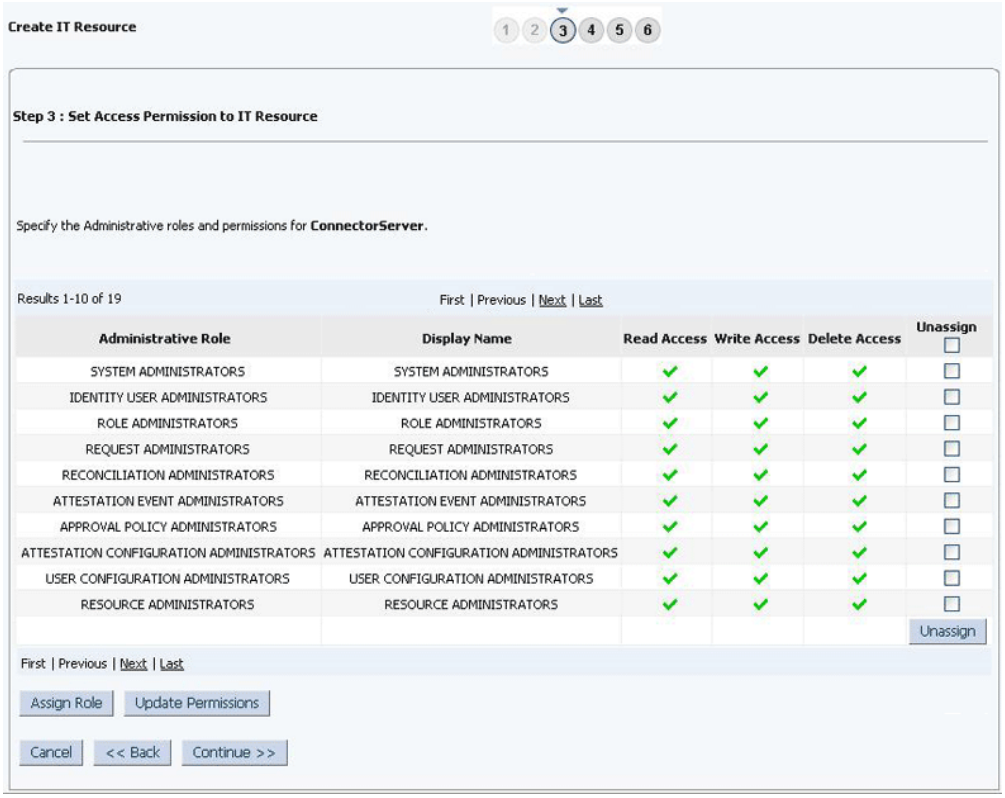
> **Note:**
>
> - This step is optional.
> - You cannot unassign the `SYSTEM ADMINISTRATORS` group. You can unassign only other groups that you assign to the IT resource.

a. Select the **Unassign** check box for the group that you want to unassign.

b. Click **Unassign.**

10. Click **Continue**. Figure 2-4 shows the Step 3: Set Access Permission to IT Resource page.

Figure 2-4 Step 3: Set Access Permission to IT Resource

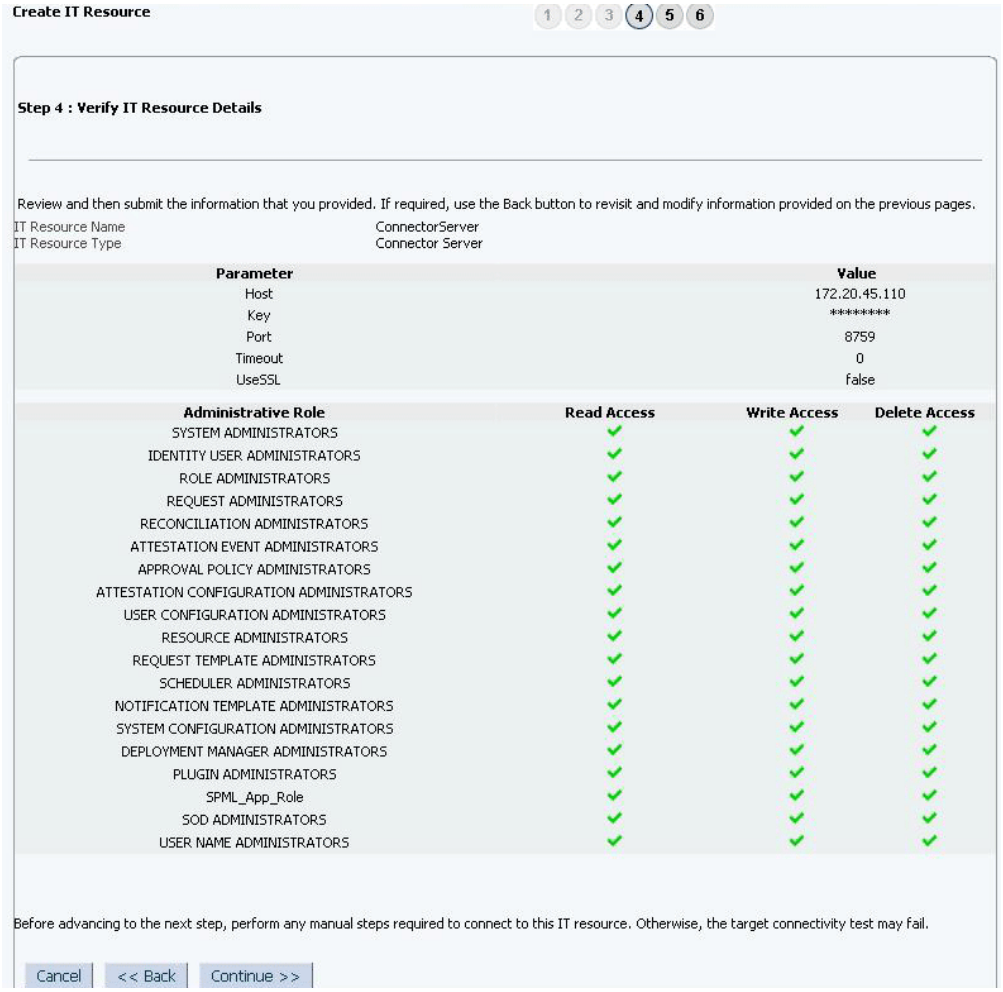


11. On the Step 4: Verify IT Resource Details page, review the information that you provided on the first, second, and third pages. If you want to make changes in the

data entered on any page, click **Back** to revisit the page and then make the required changes.

12. To proceed with the creation of the IT resource, click **Continue**. Figure 2-5 shows Step 4: Verify IT Resource Details page.

Figure 2-5 Step 4: Verify IT Resource Details



13. The Step 5: IT Resource Connection Result page displays the results of a connectivity test that is run using the IT resource information. If the test is successful, then click Continue. If the test fails, then you can perform one of the following steps:

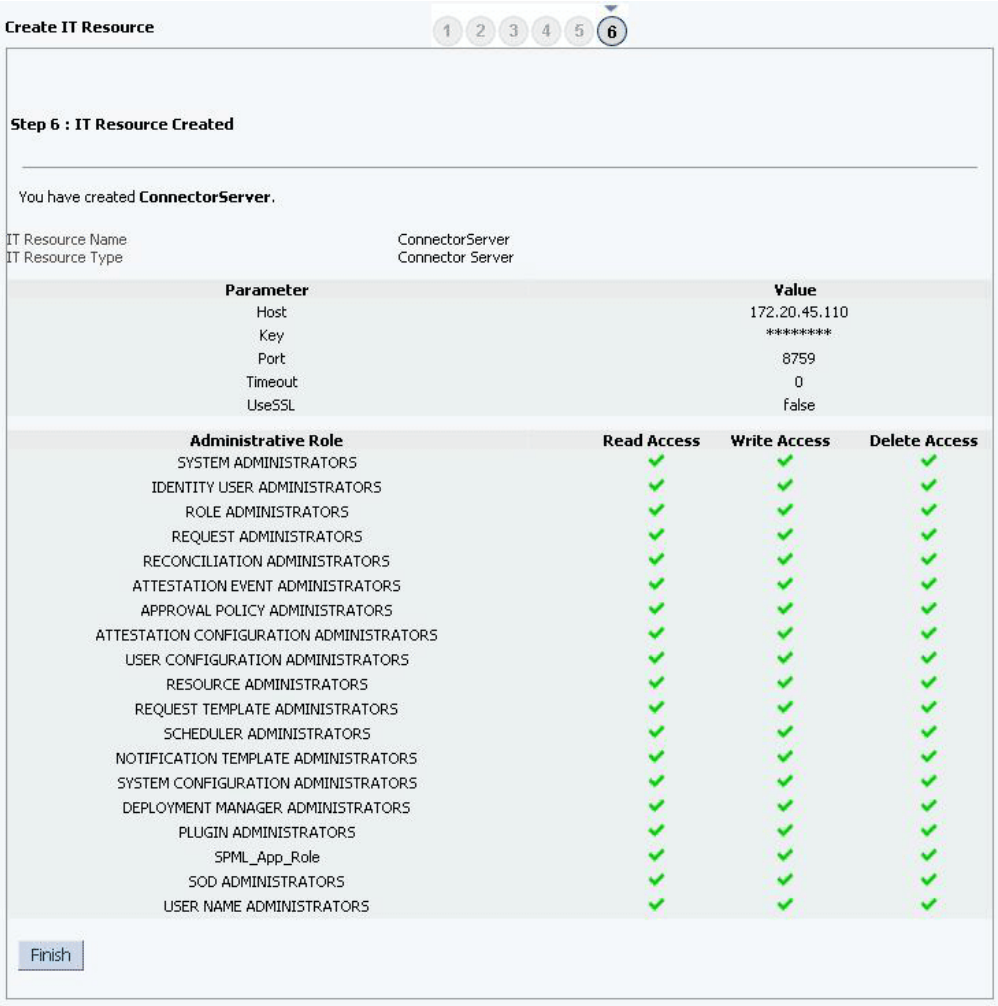
    - Click **Back** to revisit the previous pages and then make corrections in the IT resource creation information.
    - Click **Cancel** to stop the procedure, and then begin from the first step onward.

    Figure 2-6 shows the Step 5: IT Resource Connection Result page.

Figure 2-6 Step 5: IT Resource Connection Result

14. Click **Finish**. Figure 2-7 shows the IT Resource Created Page.

Figure 2-7 Step 6: IT Resource Created



2.3.8 Localizing Field Labels in UI Forms

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 (oracle.iam.console.identity.sysadmin.ear_V2.0_metadata.zip) to the local computer.
5. Extract the contents of the archive, and open one of the following files in a text editor:
   - For Oracle Identity Manager 11g Release 2 PS2 (11.1.2.2.0):

     SAVED_LOCATION\xliffBundles\oracle\iam\ui\runtime\BizEditorBundle_en.xlf
   - For releases prior to Oracle Identity Manager 11g Release 2 PS2 (11.1.2.2.0):

     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">
      ```

   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.BaseRuntimeResourceBundle']
      ['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity.userEO.UD_A
      D_USERNAME__c_description']}">
      <source>Username</source>
      </target>
      </trans-unit>
      <trans-unit
      id="sessiondef.oracle.iam.ui.runtime.form.model.googleapps.entity.googleappsE
      O.UD_GA_USR_ACCOUNT_NAME__c">
      <source>Username</source>
      </target>
      </trans-unit>
      ```

   d. Open the resource file from the connector package, for example GoogleApps_ja.properties, and get the value of the attribute from the file, for example, global.udf.UD_GA_USR_ACCOUNT_NAME=\u30A2\u30AB \u30A6\u30F3\u30C8\u540D.
   e. Replace the original code shown in Step 6.c with the following:

      ```
      <trans-unit id="$
      {adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResourceBundle']
      ['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity.userEO.UD_G
      A_USR_ACCOUNT_NAME__c_description']}">
      <source>Account Name</source>
      <target>u30A2\u30AB\u30A6\u30F3\u30C8\u540D</target>
      </trans-unit>
      <trans-unit
      id="sessiondef.oracle.iam.ui.runtime.form.model.googleapps.entity.googleappsE
      O.UD_GA_USR_ACCOUNT_NAME__c_LABEL">
      <source>Account Name</source>
      ```

```
<target>\u30A2\u30AB\u30A6\u30F3\u30C8\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.

2.4 Upgrading the Connector

If you have already deployed the Google Apps Connector versions 11.1.1.5.0 or 11.1.1.6.0, then you can upgrade the connector to version 11.1.1.7.0. The following sections discuss the procedure to upgrade the connector:

The following sections discuss the procedure to upgrade the connector:

- Preupgrade Steps
- Upgrade Steps
- Postupgrade Steps

> **Note:**
>
> Before you perform the upgrade procedure:
>
> - It is strongly recommended that you create a backup of the Oracle Identity Manager database and the connector JARs before you perform an upgrade operation. Refer to the database documentation for information about creating a backup.
> - Upgrade the Google Apps connector.
> - As a best practice, first perform the upgrade procedure in a test environment.

2.4.1 Preupgrade Steps

Preupgrade steps for the connector involves performing a reconciliation run to fetch records from the target system, defining the source connector in Oracle Identity

Manager, creating copies of the connector if you want to configure it for multiple installations of the target system, and disabling all the scheduled jobs.

Perform the following preupgrade steps:

1. Perform a reconciliation run to fetch all latest updates to Oracle Identity Manager.
2. Perform the preupgrade procedure documented in Managing Connector Lifecycle in *Oracle Fusion Middleware Administering Oracle Identity Manager*.
3. Define the source connector (an earlier release of the connector that must be upgraded) in Oracle Identity Manager. You define the source connector to update the Deployment Manager XML file with all customization changes made to the connector. See Managing Connector Lifecycle in *Oracle Fusion Middleware Administering Oracle Identity Manager* for more information.

2.4.2 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.

See Managing Connector Lifecycle in *Oracle Fusion Middleware Administering Oracle Identity Manager* for detailed information about the wizard and silent modes.

2.4.3 Postupgrade Steps

Postupgrade steps involve uploading new connector jars, running the Form Version Control (FVC) utility to manage data changes on a form, running the PostUpgradeScript.sql script to upgrade the IT resource, configuring the upgraded IT resource of the source connector, and so on.

Perform the following procedure:

1. Perform the postupgrade procedure documented in Managing Connector Lifecycle of *Oracle Fusion Middleware Administering Oracle Identity Manager*.
2. Update Oracle Identity Manager database with Google Apps connector version 11.1.1.7.0 ICF bundle JAR file, org.identityconnectors.googleapps-1.2.1.jar by following the procedure mentioned in Downloading and Copying Google Apps Third-Party Libraries.

3. If the connector is deployed on a Connector Server, then perform the procedure mentioned below:
   a. Stop the Connector Server.
   b. Replace the org.identityconnectors.googleapps-1.2.1.jar file with the Google Apps 11.1.1.7.0 connector bundle.

      > **Note:**
      >
      > If the Google Apps third party JARs are kept in an ICF bundle, then follow steps 3.a and 3.b of this procedure.

   c. Start the Connector Server.
4. If the connector is not deployed on a Connector Server, then perform the procedure mentioned below on Oracle Identity Manager:
   a. Delete the existing ICF Bundle org.identityconnectors.googleapps-1.2.1.jar from the Oracle Identity Manager database using the Delete JARs utility using option-4 which is the designated option for the ICF bundle.

      When you run the Delete JARs utility, you are prompted to enter the login credentials of the Oracle Identity Manager administrator, URL of the Oracle Identity Manager host computer, context factory value, type of JAR file being deleted, and the name of the JAR file to be removed. To delete the ICF bundle jar file, specify `4` as the value of the JAR type.

   b. Copy the ICF Bundle org.identityconnectors.googleapps-1.2.1.jar from the installation media to a local temporary folder. Create a `lib` folder in the local temporary folder created and copy the Google Apps third-party JARs in the lib folder.

      Perform the JAR file update on the ICF Bundle org.identityconnectors.googleapps-1.2.1.jar using the same "lib" folder.

      For example, jar uvf org.identityconnectors.googleapps-1.2.1.jar lib

   c. Run the Oracle Identity Manager Upload JARs utility to post the ICF bundle org.identityconnectors.googleapps-1.2.1.jar file to the Oracle Identity Manager database.

      When you run the utility, you are prompted to enter the login credentials of the Oracle Identity Manager administrator, URL of the Oracle Identity Manager host computer, context factory value, type of JAR file being uploaded, and the location from which the JAR file is to be uploaded. To upload the ICF bundle jar file, specify `4` as the value of the JAR type.
5. Run the Form Version Control (FVC) utility with the following fvc.properties for the UD_GA_USR parent form:

   ```
   ResourceObject;GoogleApps User
   FormName;UD_GA_USR
   FromVersion;<FILL_OLD_VERSION>
   ToVersion;<FILL_NEW_VERSION>
   ```

   Perform this step for child forms UD_GA_GROUP and UD_GA_NICK with their appropriate From Version and To Version values.

6. Perform the procedures mentioned in Preinstallation on the Target System and Configuring the IT Resource for the Target System.
7. Remove the mappings from the provisioning attribute map lookup definition, Lookup.GoogleApps.UM.ProvAttrMap and reconciliation attribute map lookup definition, Lookup.GoogleApps.UM.ReconAttrMap.
   - Provisioning attribute mappings:

     Sample 1:

     Code Key: `UD_GA_NICK~Nick Name`

     Decode: `nicknames`

     Sample 2:

     Code Key: `Mail Quota`

     Decode: `quota`
   - Reconciliation attribute mappings:

     Sample 1:

     Code Key: `Nick Names~Nick Name`

     Decode: `nicknames`

     Sample 2:

     Code Key: `Mail Quota`

     Decode: `quota`
8. Open the file upgrade/PostUpgradeScriptGoogleApps.sql from the installation media and replace "DOMAIN.COM" with the configured value of Google Apps IT Resource parameter "domain" and execute the script in the Oracle Identity Manager database.

   > **Note:**
   >
   > If the deployment does not have a requirement of using target user reconciliation, then the following steps can be skipped. However, in that case, updating the account name is also not supported. Hence, you must remove the "Account Name Updated Task" from the "GoogleApps User" process definition in order to stop account name updates.

9. Go to the **GoogleApps User** resource object and change "Unique Id" to Not Required from Required. Also, set the "Account Name" field to Required.

   Save the resource object.
10. Go to the **GoogleApps User** process definition and mark "Unique Id" from key field to non-key field. Also, mark "Account Name" as a key field.

    Save the resource object.
11. Go to the **GoogleApps User** resource object and select the **Create Reconciliation Profile** option to create a reconciliation profile.
12. Restart the application server running Oracle Identity Manager.

13. Run a full target user reconciliation and ensure that all the existing Google Apps accounts in Oracle Identity Manager now have "unique id" as the long value instead of the account name value.

    For example, "118384305435185484147" will be replaced with "gatest".

14. Revert the resource object and process definition changes performed in steps 7, 8, and 9 to ensure that the "Unique Id" and "IT Resource" fields are set as Key or Required reconciliation fields.

3
Using the Connector

You can use the Google Apps connector for performing reconciliation and provisioning operations after configuring it to meet your requirements.

This chapter is divided into the following sections:

> **Note:**
>
> These sections provide both conceptual and procedural information about configuring the connector. It is recommended that you read the conceptual information before you perform the procedures.

> **Note:**
>
> These sections provide both conceptual and procedural information about configuring the connector. It is recommended that you read the conceptual information before you perform the procedures.

3.1 Scheduled Task for Lookup Field Synchronization

The GoogleApps Group Lookup Reconciliation scheduled job is used for lookup field synchronization.

Values fetched by this scheduled job from the target system are populated in the Lookup.GoogleApps.Groups lookup definition. Table 3-1 describes the attributes of this scheduled job. The procedure to configure scheduled tasks is described later in this guide.

> **Note:**
>
> The target system allows you to use special characters in lookup fields. However, in Oracle Identity Manager, special characters are not supported in lookup definitions.

Table 3-1 Attributes of the GoogleApps Group Lookup Reconciliation Scheduled Job

Attribute	Description
Batch Size	Enter the number of records that must be included in each batch during reconciliation.
Code Key Attribute	This attribute holds the name of the connector attribute whose value is used to populate the Code Key column of the Lookup.GoogleApps.Groups lookup definition. Default value: `__NAME__`
Decode Attribute	This attribute holds the name of the connector attribute whose value is used to populate the Decode column of the Lookup.GoogleApps.Groups lookup definition. Default value: `__NAME__`
IT Resource Name	Name of the IT resource for the target system installation from which you reconcile user records. Default value: `GoogleApps`
Lookup Name	This attribute holds the name of the lookup definition into which values must be populated by the scheduled task. Default value: `Lookup.GoogleApps.Groups` If you create a copy of the Lookup.GoogleApps.Groups lookup definition, then enter the name of that new lookup definition as the value of the Lookup Name attribute.
Object Type	This attribute is used to perform reconciliation of specified object type. Group is the only supported object type. Default value: `Group`

3.2 Configuring Reconciliation

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

As mentioned earlier in this guide, reconciliation involves duplicating in Oracle Identity Manager the creation of and modifications to user accounts on the target system. This section discusses the following topics related to configuring reconciliation:

- Full Reconciliation
- Limited Reconciliation
- Batched Reconciliation
- Reconciliation Scheduled Jobs

3.2.1 Full Reconciliation

Full reconciliation involves reconciling all existing user records from the target system into Oracle Identity Manager.

After you deploy the connector, you must first perform full reconciliation. To perform a full reconciliation run, ensure that no value is specified for the Filter attribute of the scheduled job for reconciling users and groups.

3.2.2 Limited Reconciliation

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

The scheduled job provides a `Filter` parameter that allows you to use any of the Google Apps resource attributes to filter the target system records.

You can perform limited reconciliation by creating filters for the reconciliation module. This connector provides a Filter attribute (a scheduled task attribute) that allows you to use Google Apps resource attributes to filter the target system records.

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

While deploying the connector, follow the instructions in Configuring Scheduled Jobs to specify attribute values.

3.2.3 Batched Reconciliation

By default, all target system records that are added or modified after the last reconciliation run are reconciled during the current reconciliation run. 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 such problems.

To configure batched reconciliation, specify a value for the Batch Size attribute of the scheduled job for user and group reconciliation. You use the Batch Size attribute to specify the number of records that must be included in each batch fetched from the target system.

3.2.4 Reconciliation Scheduled Jobs

When you run the Connector Installer, reconciliation scheduled jobs are automatically created in Oracle Identity Manager. You must configure these scheduled jobs to suit your requirements by specifying values for its attributes.

You must specify values for the attributes of the following scheduled jobs:

> **Note:**
>
> Attribute values are predefined in the connector XML file that you import. Specify values only for the attributes that you want to change.

- GoogleApps Target Resource User Reconciliation
- GoogleApps Target Resource User Delete Reconciliation
- GoogleApps Group Recon
- GoogleApps Group Delete Recon

3.2.4.1 GoogleApps Target Resource User Reconciliation

You use the GoogleApps Target Resource User Reconciliation scheduled job to reconcile account data from the target system.

Table 3-2 describes the attributes of this scheduled job.

Table 3-2 Attributes of the GoogleApps Target Resource User Reconciliation Scheduled Task

Attribute	Description
Batch Size	Enter the number of records that must be included in each batch fetched from the target system.
Filter	This attribute holds the ICF Filter written using ICF-Common Groovy DSL. See Limited Reconciliation for more information about this attribute.
IT Resource Name	Enter the name of the IT resource for the target system installation from which you want to reconcile user records. Default value: `GoogleApps`
Object Type	This attribute holds the name of the object type for the reconciliation run. Default value: `User` Do not change the default value.
Resource Object Name	This attribute holds the name of the resource object used for reconciliation. Default value: `GoogleApps User` **Note:** You must not change the default value.

3.2.4.2 GoogleApps Target Resource User Delete Reconciliation

You use the GoogleApps Target Resource User Delete Reconciliation scheduled job to reconcile deleted users from the target system.

Table 3-3 describes the attributes of this scheduled job.

Table 3-3 Attributes of the GoogleApps Target Resource User Delete Reconciliation Scheduled Job

Attribute	Description
Batch Size	Enter the number of records that must be included in each batch fetched from the target system.

Table 3-3 (Cont.) Attributes of the GoogleApps Target Resource User Delete Reconciliation Scheduled Job

Attribute	Description
IT Resource Name	Enter the name of the IT resource for the target system installation from which you want to reconcile user records. Default value: `GoogleApps`
Object Type	This attribute holds the name of the object type for the reconciliation run. Default value: `User` Do not change the default value.
Resource Object Name	This attribute holds the name of the resource object used for reconciliation. Default value: `GoogleApps User`

3.2.4.3 GoogleApps Group Recon

You use the GoogleApps Group Recon scheduled job to reconcile group data from the target system.

Table 3-2 describes the attributes of this scheduled job.

Table 3-4 Attributes of the GoogleApps Group Recon Scheduled Job

Attribute	Description
Batch Size	Enter the number of records that must be included in each batch fetched from the target system.
Filter	This attribute holds the ICF Filter written using ICF-Common Groovy DSL. See Limited Reconciliation for more information about this attribute.
IT Resource Name	Enter the name of the IT resource for the target system installation from which you want to reconcile user records. Default value: `GoogleApps`
Object Type	This attribute holds the name of the object type for the reconciliation run. Default value: `Group` Do not change the default value.
Organization Name	Enter the name of the Oracle Identity Manager organization in which reconciled groups must be created or updated.
Resource Object Name	This attribute holds the name of the resource object used for reconciliation. Default value: `GoogleApps Group` **Note:** You must not change the default value.
Scheduled Task Name	Name of the scheduled task used for reconciliation. Default value: `GoogleApps Group Recon`

3.2.4.4 GoogleApps Group Delete Recon

You use the GoogleApps Group Delete Recon scheduled job to reconcile deleted groups from the target system.

Table 3-3 describes the attributes of this scheduled job.

Table 3-5 Attributes of the GoogleApps Group Delete Recon Scheduled Job

Attribute	Description
Batch Size	Enter the number of records that must be included in each batch fetched from the target system.
IT Resource Name	Enter the name of the IT resource for the target system installation from which you want to reconcile user records. Default value: `GoogleApps`
Object Type	This attribute holds the name of the object type for the reconciliation run. Default value: `Group` Do not change the default value.
Organization Name	Enter the name of the Oracle Identity Manager organization from which reconciled groups must be deleted.
Resource Object Name	This attribute holds the name of the resource object used for reconciliation. Default value: `GoogleApps Group`

3.3 Configuring Scheduled Jobs

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

This section describes the procedure to configure scheduled jobs. You can apply this procedure to configure the scheduled jobs for lookup field synchronization and reconciliation.

Table 3-6 lists the scheduled jobs that you must configure.

Table 3-6 Scheduled Jobs for Lookup Field Synchronization and Reconciliation

Scheduled Task	Description
GoogleApps Group Lookup Reconciliation	This scheduled job is used for lookup field synchronization. Scheduled Task for Lookup Field Synchronization describes this scheduled job.
GoogleApps Target Resource User Reconciliation	This scheduled job is used to fetch user data during target resource reconciliation. For information about this scheduled task and its attributes, see GoogleApps Target Resource User Reconciliation.
Google Apps Target Resource User Delete Reconciliation	This scheduled job is used to fetch data about deleted users during target resource reconciliation. During a reconciliation run, for each deleted user account on the target system, the GoogleApps resource is revoked for the corresponding OIM User. For information about this scheduled task and its attributes, see GoogleApps Target Resource User Delete Reconciliation.
GoogleApps Group Recon	This scheduled job is used to fetch data about groups during target resource reconciliation. For information about this scheduled task and its attributes, see GoogleApps Group Recon.

Table 3-6 (Cont.) Scheduled Jobs for Lookup Field Synchronization and Reconciliation

Scheduled Task	Description
GoogleApps Group Delete Recon	This scheduled job is used to reconcile data about deleted groups in the target resource mode of the connector. For information about this scheduled task and its attributes, see GoogleApps Group Delete Recon.

To configure a scheduled task:

1. If you are using Oracle Identity Manager release 11.1.1.*x:*
   a. Log in to the Administrative and User Console.
   b. On the Welcome to Oracle Identity Manager Self Service page, click **Advanced** in the upper-right corner of the page.
   c. On the Welcome to Oracle Identity Manager Advanced Administration page, in the System Management region, click **Search Scheduled Jobs**.
2. If you are using Oracle Identity Manager release 11.1.2.*x* or later:
   a. Log in to Oracle Identity System Administration.
   b. In the left pane, under System Management, click **Scheduler.**
3. Search for and open the scheduled task as follows:
   a. On the left pane, 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 cStopped status to the job.
   - **Schedule Type:** Depending on the frequency at which you want the job to run, select the appropriate schedule type.

   > **Note:**
   >
   > See Creating Jobs in *Oracle Fusion Middleware Administering Oracle Identity Manager* for detailed information about schedule types.

   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:**
>
> - Attribute values are predefined in the connector XML file that you import. Specify values only for those attributes that you want to change.
> - 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.
> - See Reconciliation Scheduled Jobs for the list of scheduled tasks and their attributes.

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

> **Note:**
>
> The Stop Execution option is available in the Oracle Administration and User console. You can use the Scheduler Status page to either start, stop, or reinitialize the scheduler.

3.4 Guidelines on Performing Provisioning Operations

The following are guidelines that you must apply while performing a provisioning operation:

- For a Create User provisioning operation, you must specify a value for the Account Name field along with the domain name. For example, `jdoe@example.com`.
- During a group provisioning operation, if you select **ANYONE_CAN_JOIN** as the value of the Who Can Join field, then you must set the value of the Allow External Members field to **True.** Before you perform the group provisioning operation with the values discussed in this point, ensure you have performed the procedure described in Preinstallation .

3.5 Performing Provisioning Operations in Oracle Identity Manager Release 11.1.1.*x*

Provisioning a resource for an OIM User involves using Oracle Identity Manager to create a Google account for the user.

When you install the connector on Oracle Identity Manager, the direct provisioning feature is automatically enabled. This means that the process form is enabled when you install the connector.

If you configure the connector for request-based provisioning, then the process form is suppressed and the object form is displayed. In other words, direct provisioning is disabled when you configure the connector for request-based provisioning. If you want to revert to direct provisioning, then perform the steps described in Switching Between Request-Based Provisioning and Direct Provisioning.

This following are types of provisioning operations:

- Direct provisioning
- Request-based provisioning

This section provides information on the following topics:

- Direct Provisioning
- Request-Based Provisioning
- Switching Between Request-Based Provisioning and Direct Provisioning

3.5.1 Direct Provisioning

You create a new user in the Administrative and User Console by using the Create User page. You provision an account for the newly created user on the Resource tab of the user details page.

To provision a resource by using the direct provisioning approach:

1. Log in to the Administrative and User Console.
2. If you want to first create an OIM User and then provision a target system account, then:
   a. On the Welcome to Identity Administration page, in the Users region, click **Create User**.
   b. On the user details page, enter values for the OIM User fields, and then click **Save**.
3. If you want to provision a target system account to an existing OIM User, then:
   a. On the Welcome to Identity Administration page, search for the OIM User by selecting Users from the list on the left pane.
   b. From the list of users displayed in the search results, select the OIM User. The user details page is displayed on the right pane.
4. On the user details page, click the **Resources** tab.
5. From the Action menu, select **Add Resource**. Alternatively, you can click the add resource icon with the plus (+) sign. The Provision Resource to User page is displayed in a new window.
6. On the Step 1: Select a Resource page, select **GoogleApps User** from the list and then click **Continue**.
7. On the Step 2: Verify Resource Selection page, click **Continue**.
8. On the Step 5: Provide Process Data for Google Users Form page, enter the details of the account that you want to create on the target system and then click **Continue.**
9. On the Step 6: Verify Process Data page, verify the data that you have provided and then click **Continue.**
10. Close the window displaying the "Provisioning has been initiated" message.
11. On the Resources tab, click **Refresh** to view the newly provisioned resource.

3.5.2 Request-Based Provisioning

A request-based provisioning operation involves both end users and approvers. Typically, these approvers are in the management chain of the requesters. The following sections discuss the steps to be performed by end users and approvers during a request-based provisioning operation:

> **Note:**
>
> The procedures described in these sections are built on an example in which the end user raises or creates a request for provisioning a target system account. This request is then approved by the approver.

- End User's Role in Request-Based Provisioning
- Approver's Role in Request-Based Provisioning

3.5.2.1 End User's Role in Request-Based Provisioning

The following steps are performed by the end user in a request-based provisioning operation:

1. Log in to the Administrative and User console.
2. On the Welcome page, click **Advanced** on the top right corner of the page.
3. On the Welcome to Identity Administration page, click the **Administration** tab, and then click the **Requests** tab.
4. From the Actions menu on the left pane, select **Create Request**.

   The Select Request Template page is displayed.
5. From the Request Template list, select **Provision Resource** and click **Next**.
6. On the Select Users page, specify a search criterion in the fields to search for the user that you want to provision the resource, and then click **Search**. A list of users that match the search criterion you specify is displayed in the Available Users list.
7. From the Available Users list, select the user to whom you want to provision the account.

   If you want to create a provisioning request for more than one user, then from the Available Users list, select users to whom you want to provision the account.
8. Click **Move** or **Move All** to include your selection in the Selected Users list, and then click **Next**.
9. On the Select Resources page, click the arrow button next to the Resource Name field to display the list of all available resources.
10. From the Available Resources list, select **GoogleApps User**, move it to the Selected Resources list, and then click **Next**.
11. On the Resource Details page, enter details of the account that must be created on the target system, and then click **Next**.

12. On the Justification page, you can specify values for the following fields, and then click **Finish**.
    - Effective Date
    - Justification

    A message confirming that your request has been sent successfully is displayed along with the Request ID.
13. If you click the request ID, then the Request Details page is displayed.
14. To view details of the approval, on the Request Details page, click the **Request History** tab.

3.5.2.2 Approver's Role in Request-Based Provisioning

The following are steps performed by the approver in a request-based provisioning operation:

The following are steps that the approver can perform:

1. Log in to the Administrative and User console.
2. On the Welcome page, click **Self-Service** in the upper-right corner of the page.
3. On the Welcome to Identity Manager Self Service page, click the **Tasks** tab.
4. On the Approvals tab, in the first section, you can specify a search criterion for request task that is assigned to you.
5. From the search results table, select the row containing the request you want to approve, and then click **Approve Task**.

   A message confirming that the task was approved is displayed.

3.5.3 Switching Between Request-Based Provisioning and Direct Provisioning

> **Note:**
>
> It is assumed that you have performed the procedure described in Enabling Request-Based Provisioning.

The following sections provide information on switching between request-based provisioning and direct provisioning:

- Switching From Request-Based Provisioning to Direct Provisioning
- Switching From Direct Provisioning to Request-Based Provisioning

3.5.3.1 Switching From Request-Based Provisioning to Direct Provisioning

To do so:

1. Log in to the Design Console.
2. Disable the Auto Save Form feature as follows:

a. Expand **Process Management**, and then double-click **Process Definition**.
b. Search for and open the **GoogleApps User** process definition.
c. Deselect the **Auto Save Form** check box.
d. Click **Save**.

3. If the Self Request Allowed feature is enabled, then:
a. Expand **Resource Management**, and then double-click **Resource Objects**.
b. Search for and open the **GoogleApps User** resource object.
c. Deselect the **Self Request Allowed** check box.
d. Click **Save**.

3.5.3.2 Switching From Direct Provisioning to Request-Based Provisioning

To do so:

1. Log in to the Design Console.
2. Enable the Auto Save Form feature as follows:
a. Expand **Process Management**, and then double-click **Process Definition**.
b. Search for and open the **GoogleApps User** process definition.
c. Select the **Auto Save Form** check box.
d. Click **Save**.
3. If you want to enable end users to raise requests for themselves, then:
a. Expand **Resource Management**, and then double-click **Resource Objects**.
b. Search for and open the **GoogleApps User** resource object.
c. Select the Self Request Allowed check box.
d. Click **Save**.

3.6 Performing Provisioning Operations in Oracle Identity Manager Release 11.1.2.*x* or Later

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 Manager release 11.1.2.*x* or later:

1. Log in to Identity Self Service.
2. Create a user. See Managing Users in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Manager* for more information about creating a user.
3. On the Account tab, click **Request Accounts.**
4. In the Catalog page, search for and add to cart the application instance created in Step 3, and then click **Checkout.**

5. Specify value for fields in the application form and then click **Ready to Submit.**
6. Click **Submit.**
7. If you want to provision entitlements, then:
    a. On the Entitlements tab, click **Request Entitlements.**
    b. In the Catalog page, search for and add to cart the entitlement, and then click **Checkout.**
    c. Click **Submit.**

3.7 Uninstalling the Connector

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

If you want to uninstall the connector for any reason, see Uninstalling Connectors in *Oracle Fusion Middleware Administering Oracle Identity Manager*.

4

Extending the Functionality of the Connector

You can extend the functionality of the connector to address your specific business requirements.
This chapter contains the following topics:

> **Note:**
>
> From Oracle Identity Manager Release 11.1.2 onward, lookup queries are not supported. See Managing Lookups of *Oracle Fusion Middleware Administering Oracle Identity Manager* guide for information about managing lookups by using the Form Designer in Identity System Administration.

4.1 Configuring Validation of Data During Reconciliation and Provisioning

You can configure validation of reconciled and provisioned single-valued data according to your requirements.

For example, you can validate data fetched from the Given Name attribute to ensure that it does not contain the number sign (#). In addition, you can validate data entered in the Given Name field on the process form so that the number sign (#) is not sent to the target system during provisioning operations.

For data that fails the validation check, the following message is displayed or recorded in the log file:

`Validation failed for attribute ATTRIBUTE_NAME.`

> **Note:**
>
> This feature cannot be applied to the Locked/Unlocked status attribute of the target system.

To configure validation of data:

1. Write code that implements the required validation logic in a Java class.

   The validation class must implement validate method with the following method signature:

   ```
   boolean validate(HashMap hmUserDetails, HashMap hmEntitlementDetails, String
   field)
   ```

   The following sample validation class checks if the value in the Given Name attribute contains the number sign (#):

   ```
   public boolean validate(HashMap hmUserDetails,
                 HashMap hmEntitlementDetails, String field) {
               /*
            * You must write code to validate attributes. Parent
            * data values can be fetched by using hmUserDetails.get(field)
            * For child data values, loop through the
            * ArrayList/Vector fetched by hmEntitlementDetails.get("Child Table")
            * Depending on the outcome of the validation operation,
            * the code must return true or false.
            */
            /*
            * In this sample code, the value "false" is returned if the field
            * contains the number sign (#). Otherwise, the value "true" is
            * returned.
            */
               boolean valid=true;
               String sGivenName=(String) hmUserDetails.get(field);
               for(int i=0;i<sGivenName.length();i++){
                 if (sGivenName.charAt(i) == '#'){
                       valid=false;
                       break;
                 }
               }
               return valid;
         }
   ```

2. Create a JAR file to hold the Java class.

3. Copy the JAR file to Oracle Identity Manager database.

   Run the Oracle Identity Manager Upload JARs utility to post the JAR file to the Oracle Identity Manager database. This utility is copied into the following location when you install Oracle Identity Manager:

   > **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 Manager administrator, URL of the Oracle Identity Manager host computer, context factory value, type of JAR file being uploaded, and the location from which the JAR file is to be uploaded. Specify 1 as the value of the JAR type.

4. If you created the Java class for validating a process form field for reconciliation, then:
    a. Log in to the Design Console.
    b. Create a lookup definition named **Lookup.GoogleApps.UM.ReconValidation.**
    c. In the Code Key, enter the resource object field name that you want to validate. For example, `Given Name`. In the Decode, enter the class name. For example, `org.identityconnectors.googleapps.extension.GAPPSValidator`.
    d. Save the changes to the lookup definition.
    e. Search for and open the **Lookup.GoogleApps.UM.Configuration** lookup definition.
    f. In the Code Key column, enter `Recon Validation Lookup`. In the Decode column, enter `Lookup.GoogleApps.UM.ReconValidation`.
    g. Save the changes to the lookup definition.
5. If you created the Java class for validating a process form field for provisioning, then:
    a. Log in to the Design Console.
    b. Create a lookup definition named **Lookup.GoogleApps.UM.ProvValidation**.
    c. In the Code Key column, enter the process form field name. For example, `Given Name`. In the Decode column, enter the class name. For example, `org.identityconnectors.googleapps.extension.GAPPSValidator`.
    d. Save the changes to the lookup definition.
    e. Search for and open the **Lookup.GoogleApps.UM.Configuration** lookup definition.
    f. In the Code Key column, enter `Provisioning Validation Lookup`. In the Decode column, enter `Lookup.GoogleApps.UM.ProvValidation`.
    g. Save the changes to the lookup definition.

4.2 Configuring Transformation of Data During User Reconciliation

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

> **Note:**
>
> This feature cannot be applied to the Locked/Unlocked status attribute of the target system.

To configure transformation of single-valued account data fetched during reconciliation:

1. Write code that implements the required transformation logic in a Java class.

   The transformation class must implement the transform method with the following method signature:

   ```
   Object transform(HashMap hmUserDetails, HashMap hmEntitlementDetails, String
   sField)
   ```

   The following sample transformation class creates a value for the Full Name attribute by using values fetched from the Given Name and Family Name attributes of the target system:

   ```
   package oracle.iam.connectors.common.transform;

   import java.util.HashMap;

   public class TransformAttribute {

         /*
         Description:Abstract method for transforming the attributes

         param hmUserDetails<String,Object>

         HashMap containing parent data details

         param hmEntitlementDetails <String,Object>

         HashMap containing child data details

         */
         public Object transform(HashMap hmUserDetails, HashMap
   hmEntitlementDetails,String sField) {
         /*
          * You must write code to transform the attributes.
          Parent data attribute values can be fetched by
          using hmUserDetails.get("Field Name").
          *To fetch child data values, loop through the
          * ArrayList/Vector fetched by hmEntitlementDetails.get("Child Table")
          * Return the transformed attribute.
          */
         String sGivenName= (String)hmUserDetails.get("Given Name");
         String sFamilyName= (String)hmUserDetails.get("Family Name");
         String sFullName=sGivenName+"."+sFamilyName;
         return sFullName;
         }
   }
   ```

2. Create a JAR file to hold the Java class.
3. Copy the JAR file to Oracle Identity Manager database.

   Run the Oracle Identity Manager Upload JARs utility to post the JAR file to the Oracle Identity Manager database. This utility is copied into the following location when you install Oracle Identity Manager:

   > **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 Manager administrator, URL of the Oracle Identity Manager host computer, context factory value, type of JAR file being uploaded, and the location from which the JAR file is to be uploaded. Specify 1 as the value of the JAR type.

4. If you created the Java class for transforming a process form field for reconciliation, then:
   a. Log in to the Design Console.
   b. Create a lookup definition named **Lookup.GoogleApps.UM.ReconTransformation**.
   c. In the Code Key column, enter the resource object field name on which you want to apply transformation. For example, `Given Name.` In the Decode column, enter the name of the class that implements the transformation logic. For example, `oracle.iam.connectors.common.transform.TransformAttribute.`
   d. Save the changes to the lookup definition.
5. Add an entry in the Lookup.GoogleApps.UM.Configuration lookup definition to enable transformation as follows:
   a. Expand Administration, and then double-click **Lookup Definition**.
   b. Search for and open the **Lookup.GoogleApps.UM.Configuration** lookup definition.
   c. In the Code Key column, enter `Recon Transformation Lookup.` In the Decode column, enter `Lookup.GoogleApps.UM.ReconTransformation.`
   d. Save the changes to the lookup definition.

4.3 Configuring the Connector for Multiple Installations of the Target System

You might want to configure the connector 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. The company has recently installed Oracle Identity Manager, and they want to configure Oracle Identity Manager to link all the installations of the target system.

To meet the requirement posed by such a scenario, you must create copies of the connector. See Cloning Connectors in *Oracle Fusion Middleware Administering Oracle Identity Manager* for more information.

4.4 Defining the Connector

Defining a connector is equivalent to registering the connector with Oracle Identity Manager. You can define a customized or reconfigured connector using Oracle Identity System Administration. After you define a connector, a record representing the connector is created in the Oracle Identity Manager database.

A connector is automatically defined when you install it using the Install Connectors feature or when you upgrade it using the Upgrade Connectors feature. You must manually define a connector if:

- You import the connector by using the Deployment Manager.
- You customize or reconfigure the connector.
- You upgrade Oracle Identity Manager.

The following events take place when you define a connector:

- A record representing the connector is created in the Oracle Identity Manager database. If this record already exists, then it is updated:
- The status of the newly defined connector is set to Active. In addition, the status of a previously installed release of the same connector automatically is set to Inactive.

See Defining Connectors in *Oracle Fusion Middleware Administering Oracle Identity Manager* for detailed information.

5

Known Issues, Workarounds, and Troubleshooting

This chapter discusses the following topics related to connector testing:

- Known Issues and Workarounds
- Troubleshooting

5.1 Known Issues and Workarounds

There are no known issues associated with this release of the connector.

5.2 Troubleshooting

Table 5-1 lists solutions to some commonly encountered issues associated with the Google Apps connector.

Table 5-1 Troubleshooting

Problem	Solution
The following javax.net.ssl.SSLKeyException occurs during reconciliation and provisioning: `javax.net.ssl.SSLKeyException: [Security:090504]Certificate chain received from www-proxy.example.com - 148.87.19.20 --> apps-apis.google.com failed hostname verification check. Certificate contained *.google.com but check expected apps-apis.google.com javax.net.ssl.SSLKeyException: [Security:090504]Certificate chain received from www-proxy.example.com - 148.87.19.20 --> apps-apis.google.com failed hostname verification check. Certificate contained *.google.com but check expected apps-apis.google.com`	If Oracle Identity Manager is deployed on WebLogic application server with Host Name Verification feature enabled, then you can disable it or use the Custom Host Name Verification feature. However, it is recommended to use Custom Host Name Verification for production environments. See the "Using Host Name Verification" section of *BEA WebLogic Server 8.1 Documentation* for more details.

A

Files and Directories in the Google Apps Connector Installation Package

This appendix provides the list of files and directories in the connector installation package and their descriptions.

Table A-1 Files and Directories In the Installation Package

File in the Installation Packages	Description
bundle/ org.identityconnectors.googleapps-1.2.1.jar	This JAR is the Identity Connector bundle. See Connector Architecture for information about this bundle.
configuration/GoogleApps-CI.xml	This XML file contains configuration information that is used during connector installation.
Files in the dataset directory ModifyProvisionedResource_GoogleAppsUser.xml ProvisionResource_GoogleAppsUser.xml	These XML files specify the information to be submitted by the requester during a request-based provisioning operation. **Note:** These files will *not* be used if you are using Oracle Identity Manager release 11.1.2.*x* or later.
resources/ GoogleApps.properties	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 Manager database. **Note:** A resource bundle is a file containing localized versions of the text strings that include GUI element labels and messages.
xml/Google Apps-ConnectorConfig.xml	This XML file contains definitions for the following connector objects: • IT resource definition • Process form • Lookup definitions • Resource object • Process definition • Scheduled tasks
xml/GoogleApps-Datasets.xml	This XML file contains the dataset file in DM format. **Note:** This dataset must *not* be imported if you are using Oracle Identity Manager release 11.1.2.*x* or later.