Oracle® Identity Governance

Configuring the Eloqua Application

12c (12.2.1.3.0)
F26417-02
October 2020

Oracle Identity Governance Configuring the Eloqua Application, 12c (12.2.1.3.0)

F26417-02

Primary Author: Gowri.G.R

Contributors: Raghunath Edhara

Contents

Preface

What’s New in This Guide?

1 About the Connector

2 Creating an Application by Using the Connector

3 Configuring the Connector

4 Installing the Connector in CI Mode

5 Using the Connector

List of Figures

List of Tables

Preface

This guide describes the connector that is used to onboard Eloqua application 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

Conventions

The following text conventions are used in this document:

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

What’s New in This Guide?

These are the updates made to the software and documentation for release 12.2.1.3.0.

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.

Software Updates in Release 12.2.1.3.0

The following is a software update in release 12.2.1.3.0:

Support for Onboarding Applications Using the Connector

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

Documentation-Specific Updates

These are the updates made to the connector documentation.

Documentation-Specific Updates in Release 12.2.1.3.0

The following documentation-specific update has been made in revision "02" of this guide:

Logger names present in Enabling Logging have been updated.

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

This is the first release of this connector. Therefore, there are no documentation-specific updates in this release.

1 About the Connector

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

The Eloqua connector lets you create and onboard Eloqua applications in Oracle Identity Governance.

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

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

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

The following topics provide a high-level overview of the Eloqua connector:

- Certified Components
- Usage Recommendation
- Certified Languages
- Supported Connector Operations
- Connector Architecture
- Use Cases Supported by the Connector
- Connector Features

1.1 Certified Components

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

Table 1-1 Certified Components

Component	Requirement for AOB Application	Requirement for CI-Based Connector
Oracle Identity Governance or Oracle Identity Manager	You can use one of the following releases: • Oracle Identity Governance release 12*c* PS4 (12.2.1.4.0) • Oracle Identity Governance 12*c* (12.2.1.3.0) **Note**: Ensure that you download and apply patches 26616250 and 25323654 from My Oracle Support. Failing to apply these patches will prevent you from successfully testing connection between Oracle Identity Governance and your target system.	You can use one of the following releases of Oracle Identity Governance or Oracle Identity Manager: • Oracle Identity Governance release 12*c* PS4 (12.2.1.4.0) • Oracle Identity Governance 12*c* (12.2.1.3.0)
Target systems	Eloqua	Eloqua
Connector Server	11.1.2.1.0 or later	11.1.2.1.0 or later
Connector Server JDK	JDK 1.8 and later	JDK 1.8 and later

1.2 Usage Recommendation

This is the recommendation for the Eloqua connector version that you can deploy and use depending on the Oracle Identity Governance or Oracle Identity Manager version that you are using.

If you are using Oracle Identity Governance 12c (12.2.1.3.0), then use the latest 12.2.1.*x* version of this connector. Deploy the connector using the **Applications** option on the **Manage** tab of Identity Self Service.

1.3 Certified Languages

These are the languages that the connector supports.

- Arabic
- Chinese (Simplified)
- Chinese (Traditional)
- Czech
- Danish
- Dutch
- English (US)
- Finnish

- French
- French (Canadian)
- German
- Greek
- Hebrew
- Hungarian
- Italian
- Japanese
- Korean
- Norwegian
- Polish
- Portuguese
- Portuguese (Brazilian)
- Romanian
- Russian
- Slovak
- Spanish
- Swedish
- Thai
- Turkish

1.4 Supported Connector Operations

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

Table 1-2 Supported Connector Operations

Operation	Supported
User Management	
Create user	Yes
Update user	Yes
Delete user	Yes
Reset Password	Yes
License Grant Management	
Grant and Revoke Licences	Yes
Group Management	
Add and Remove Groups	Yes

Note:

All the connector artifacts required for managing groups as an object (for example groups attribute mappings, reconciliation rules, jobs, and so on) are not visible in the Applications UI in Identity Self Service. However, all the required information is available in the predefined application templates of the connector installation package.

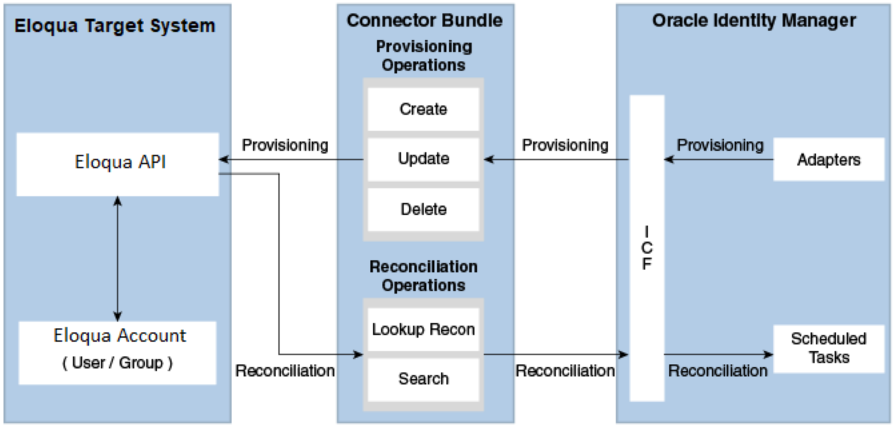
1.5 Connector Architecture

The Eloqua connector is implemented by using the Identity Connector Framework (ICF).

The ICF is a component that is required in order to use Identity Connector. ICF provides basic reconciliation and provisioning operations that are common to all Oracle Identity Governance connectors. In addition, ICF provides common features that developers would otherwise need to implement on their own, such as, buffering, time outs, and filtering. ICF is distributed together with Oracle Identity Governance. Therefore, you do not need to configure or modify ICF.

Figure 1-1 shows the architecture of the Eloqua connector.

Figure 1-1 Connector Architecture



The connector is configured to run in the account management mode. Account management is also known as target resource management. In this mode, the target system is used as a target resource and the connector enables the following operations:

- Provisioning

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

- Target resource reconciliation

  In target resource reconciliation, data related to the newly created and modified target system accounts can be reconciled and linked with existing OIM Users and provisioned resources. You use a scheduled job for performing reconciliation.

The Eloqua Identity Connector Bundle communicates with the Eloqua API using the HTTPS protocol. The Eloqua API provides programmatic access through REST API endpoints. Apps can use the Eloqua API to perform create, read, update, and delete (CRUD) operations on directory data and directory objects, such as users.

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

1.6 Use Cases Supported by the Connector

The Eloqua connector is used to integrate Oracle Identity Governance with Eloqua to ensure that all Eloqua accounts are created, updated, and deactivated on an integrated cycle with the rest of the identity-aware applications in your enterprise. The Eloqua connector supports management of identities for Cloud Identity, Synchronized Identity, and Federated Identity models of Eloqua. In a typical IT scenario, an organization using Oracle Identity Governance wants to manage accounts, groups, and licenses across Eloqua Cloud Service.

The following are some of the most common scenarios in which this connector can be used:

- **Eloqua User Management**

  An organization using Eloqua wants to integrate with Oracle Identity Governance to manage identities. The organization wants to manage its user identities by creating them in the target system using Oracle Identity Governance. The organization also wants to synchronize user identity changes performed directly in the target system with Oracle Identity Governance. In such a scenario, a quick and an easy way is to install the Eloqua connector and configure it with your target system by providing connection information.

  To create a new user in the target system, fill in and submit the OIM process form to trigger the provisioning operation. The connector executes the CreateOp operation against your target system and the user is created on successful execution of the operation. Similarly, operations like delete and update can be performed.

  To search or retrieve the user identities, you must run a scheduled task from Oracle Identity Governance. The connector will run the corresponding SearchOp against the user identities in the target system and fetch all the changes to Oracle Identity Governance.

- **Eloqua Group Management**

  An organization has a number of Eloqua Security Groups allowing its users to assign and unassign groups. By using the Eloqua connector, you can effectively track all user groups by leveraging Oracle Identity Governance capability.

- **Eloqua User License Management**

  Another scenario is one in which an organization is using Eloqua for business and manages user licenses as per the changing needs of the organization by assigning or unassigning licenses for users. What is needed is an effective way to keep track of all the licenses and user rights both in cloud and on-premise servers. In such a scenario, you can use the Eloqua connector to effectively track all user licenses. You can keep track of these license assignment changes by leveraging Oracle Identity Governance capability of auditing and reporting.

1.7 Connector Features

The features of the connector include support for connector server, full reconciliation, limited reconciliation, and reconciliation of deleted account data.

Table 1-3 provides the list of features supported by the AOB application and CI-based connector.

Table 1-3 Supported Connector Features Matrix

Feature	AOB Application	CI-Based Connector
Full reconciliation	Yes	Yes
Incremental reconciliation	Yes	Yes
Limited reconciliation	Yes	Yes
Delete reconciliation	Yes	Yes
Use connector server	Yes	Yes
Transformation and validation of account data	Yes	Yes
Perform connector operations in multiple domains	Yes	Yes
Support for paging	Yes	Yes
Test connection	Yes	No

The following topics provide more information on the features of the AOB application:

- Full Reconciliation and Incremental Reconciliation
- Support for the Connector Server
- Limited Reconciliation
- Transformation and Validation of Account Data

1.7.1 Full Reconciliation and Incremental Reconciliation

You can perform full reconciliation to bring all existing user data from the target system to Oracle Identity Governance.

After the first full reconciliation run, you can configure your connector for incremental reconciliation if the target system contains an attribute that holds the timestamp at which an object is created or modified.

In incremental reconciliation, only records that are added or modified after the last reconciliation run are fetched into Oracle Identity Governance. During an incremental reconciliation run, the scheduled job fetches only target system records that are added or modified after the time-stamp stored in the Latest Token attribute of the scheduled job.

> **Note:**
>
> The connector supports incremental reconciliation if the target system contains an attribute that holds the timestamp at which an object is created or modified.

You can perform a full reconciliation run at any time. See Performing Full and Incremental Reconciliation for more information about performing full and incremental reconciliation.

1.7.2 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 more information about installing and configuring connector server and running the connector server

1.7.3 Limited Reconciliation

You can reconcile records from the target system based on a specified filter criterion. To limit or filter the records that are fetched into Oracle Identity Governance during a reconciliation run, you can specify the subset of added or modified target system records that must be reconciled.

You can set a reconciliation filter as the value of the Filter Suffix attribute of the user reconciliation scheduled job. The Filter Suffix attribute helps you to assign filters to the API based on which you get a filtered response from the target system.

For more information, see Performing Limited Reconciliation.

1.7.4 Transformation and Validation of Account Data

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

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

2

Creating an Application by Using the Connector

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

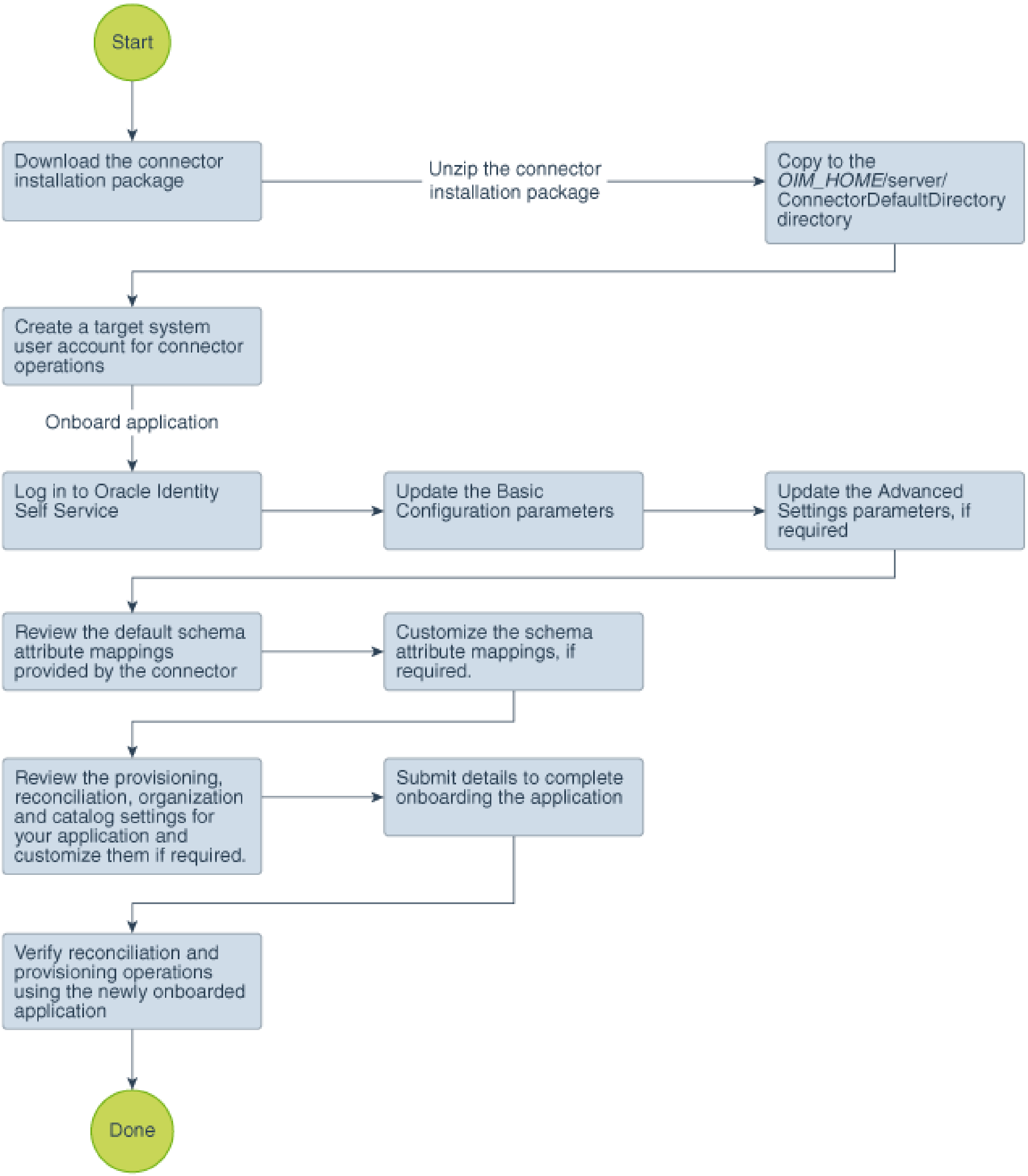
- Process Flow for Creating an Application By Using the Connector
- Downloading the Connector Installation Package
- Creating an Application By Using the Connector

2.1 Process Flow for Creating an Application By Using the Connector

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

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

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


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

2.2 Downloading the Connector Installation Package

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

To download the connector installation package:

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

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

2.3 Creating an Application By Using the Connector

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

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

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

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

The application is created in Oracle Identity Governance.

h. When you are prompted whether you want to create a default request form, click **Yes** or **No**.

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

2. Verify reconciliation and provisioning operations on the newly created application.

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

3
Configuring the Connector

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

- Basic Configuration Parameters
- Advanced Settings Parameters
- Attribute Mappings
- Correlation Rules, Situations, and Responses for a Target Application
- Reconciliation Jobs

3.1 Basic Configuration Parameters

These are the connection-related parameters that Oracle Identity Governance requires to connect to an Eloqua application.

> **Note:**
>
> Unless specified, do not modify entries in the below table.

Table 3-1 Parameters in the Basic Configuration

Parameter	Mandatory ?	Description
authenticationType	Yes	Enter the type of authentication used by your target system. **Sample value**: `basic`
username	Yes	Enter the user name of the target system that you create for performing connector operations. **Sample value**: `IDMsysE10OD01\johnsmith`
password	Yes	Enter the password of the target system user account that you create for connector operations. **Sample value**: `password`

Table 3-1 (Cont.) Parameters in the Basic Configuration

Parameter	Mandatory ?	Description
host	Yes	Enter the host name of the machine hosting your target system. This is a mandatory attribute while creating an application. **Sample value**: `www.example.com`
port	No	Enter the port number at which the target system is listening. **Sample value**: `443`
proxyHost	No	Enter the name of the proxy host used to connect to an external target. **Sample value**: `www.example.com`
proxyPassword	No	Enter the password of the proxy user ID of the target system user account that Oracle Identity Governance uses to connect to the target system.
proxyPort	No	Enter the proxy port number. **Sample value**: `80`
proxyUser	No	Enter the proxy user name of the target system user account that Oracle Identity Governance uses to connect to the target system.
sslEnabled	No	If the target system requires SSL connectivity, then set the value of this parameter to `true`. Otherwise set the value to `false`. **Default value:** `true`

3.2 Advanced Settings Parameters

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

> **Note:**
>
> - Unless specified, do not modify entries in the below table.
> - All parameters in the below table are mandatory.

Table 3-2 Advanced Settings Parameters

Parameter	Description
relURIs	This entry holds the relative URL of every object class supported by this connector and the connector operations that can be performed on these object classes. This is a mandatory attribute while creating an application. **Default value**: `"__ACCOUNT__.CREATEOP=/API/REST/2.0/system/user","__ACCOUNT__.UPDATEOP=/API/REST/2.0/system/user/$(__UID__)$","__ACCOUNT__.SEARCHOP=/API/REST/2.0/system/users?depth=Partial&$(Filter Suffix)$&page=$(PAGE_INCREMENT)$&count=$(PAGE_SIZE)$","__ACCOUNT__.DELETEOP=/API/REST/2.0/system/user/$(__UID__)$","__ACCOUNT__.__ENABLE__.UPDATEOP=/API/REST/2.0/system/user/$(__UID__)$/enabled","__ACCOUNT__.securityGroups.REMOVEATTRIBUTE=/api/REST/2.0/system/security/group/$(securityGroups.id)$/users","__ACCOUNT__.securityGroups.ADDATTRIBUTE=/api/REST/2.0/system/security/group/$(securityGroups.id)$/users","securityGroups.SEARCHOP=/api/REST/2.0/system/security/groups","license.SEARCHOP=/api/REST/2.0/system/security/licenses","__ACCOUNT__.license.SEARCHOP=/api/REST/2.0/system/user/$(__UID__)$/security/licenses","__ACCOUNT__.license.ADDATTRIBUTE=/api/REST/2.0/system/user/$(__UID__)$/security/licenses","__ACCOUNT__.license.REMOVEATTRIBUTE=/api/REST/2.0/system/user/$(__UID__)$/security/licenses","__ACCOUNT__.__PASSWORD__.CREATEOP=/API/REST/2.0/system/user/$(__UID__)$/password","__ACCOUNT__.__PASSWORD__.UPDATEOP=/API/REST/2.0/system/user/$(__UID__)$/password"`

Table 3-2 (Cont.) Advanced Settings Parameters

Parameter	Description
nameAttributes	This entry holds the name attribute for all the objects that are handled by this connector. For example, for the `__ACCOUNT__` object class that it used for User accounts, the name attribute is `userPrincipalName`. **Default value**: `"__ACCOUNT__.loginName","securityGroups.id","license.name"`
uidAttributes	This entry holds the uid attribute for all the objects that are handled by this connector. **Default value**:`"__ACCOUNT__.id","securityGroups.name","license.code"`
Bundle Name	This entry holds the name of the connector bundle. **Default value**: `org.identityconnectors.genericrest`
Bundle Version	This entry holds the version of the connector bundle. **Default value**: `12.3.0`
Connector Name	This entry holds the name of the connector class. **Default value**: `org.identityconnectors.genericrest.GenericRESTConnector`
opTypes	This entry specifies the HTTP operation type for each object class supported by the connector. Values are comma separated and are in the following format: *OBJ_CLASS.OP=HTTP_OP* In this format, *`OBJ_CLASS`* is the connector object class, `OP` is the connector operation (for example, CreateOp, UpdateOp, SearchOp), and `HTTP_OP` is the HTTP operation (GET, PUT, or POST). **Default value**: `"__ACCOUNT__.CREATEOP=POST","__ACCOUNT__.UPDATEOP=PUT","__ACCOUNT__.__ENABLE__.UPDATEOP=PUT","__ACCOUNT__.SEARCHOP=GET","__ACCOUNT__.TESTOP=GET","__ACCOUNT__.DELETEOP=DELETE","__ACCOUNT__.securityGroups.ADDATTRIBUTE=PATCH","__ACCOUNT__.securityGroups.REMOVEATTRIBUTE=PATCH","__ACCOUNT__.license.ADDATTRIBUTE=PATCH","__ACCOUNT__.license.REMOVEATTRIBUTE=PATCH","__ACCOUNT__.__PASSWORD__.CREATEOP=PUT","__ACCOUNT__.__PASSWORD__.UPDATEOP=PUT"`

Table 3-2 (Cont.) Advanced Settings Parameters

Parameter	Description
Any Incremental Recon Attribute Type	By default, during incremental reconciliation, Oracle Identity Governance accepts timestamp information sent from the target system only in Long datatype format. Setting the value of this parameter to `True` indicates that Oracle Identity Governance will accept timestamp information in any datatype format. **Default value**: `True`
jsonResourcesTag	This entry holds the json tag value that is used during reconciliation for parsing multiple entries in a single payload. **Default value**: `"__ACCOUNT__=elements","securityGroups=elements","license=elements","__ACCOUNT__.license=elements"`
httpHeaderContentType	This entry holds the content type expected by the target system in the header. **Default value**: `application/json`
httpHeaderAccept	This entry holds the accept type expected from the target system in the header. **Default value**: `application/json`
specialAttributeTargetFormat	This entry lists the format in which an attribute is present in the target system endpoint. Values are comma separated and are presented in the following format: *OBJ_CLASS.ATTR_NAME=TARGET_FORMAT* **Default value**`"__ACCOUNT__.securityGroups=elements.id"`
specialAttributeHandling	This entry lists the special attributes whose values should be sent to the target system one by one ("SINGLE"). Values are comma separated and are in the following format: *OBJ_CLASS.ATTR_NAME.PROV_OP*=SINGLE For example, the `__ACCOUNT__.manager.UPDATEOP=SINGLE` value in decode implies that during an update provisioning operation, the `manager` attribute of the `__ACCOUNT__` object class must be sent to the target system one-by-one. **Default value**`"__ACCOUNT__.securityGroups.ADDATTRIBUTE=SINGLE","__ACCOUNT__.securityGroups.REMOVEATTRIBUTE=SINGLE","__ACCOUNT__.license.ADDATTRIBUTE=SINGLE","__ACCOUNT__.license.REMOVEATTRIBUTE=SINGLE"`

Table 3-2 (Cont.) Advanced Settings Parameters

Parameter	Description
pageSize	This entry holds how many resources appears on a page for a search operation. **Default value:** `example 5`
customPayload	This entry lists the payloads for all operations that are not in the standard format. **Default value:**`"__ACCOUNT__.__ENABLE__.UPDATEOP={\"Enabled\": \"$(__ENABLE__)$\"}","__ACCOUNT__.securityGroups.ADDATTRIBUTE=[{\"type\": \"SecurityGroupUser\",\"patchMethod\": \"add\",\"user\": {\"type\": \"User\",\"id\": \"$(__UID__)$\"}}]","__ACCOUNT__.securityGroups.REMOVEATTRIBUTE=[{\"type\": \"SecurityGroupUser\",\"patchMethod\": \"remove\",\"user\": {\"type\": \"User\",\"id\": \"$(__UID__)$\"}}]","__ACCOUNT__.license.ADDATTRIBUTE=[{\"type\": \"LicenseGrant\",\"patchMethod\": \"add\",\"license\": {\"type\": \"License\",\"code\": \"$(code)$\"}}]","__ACCOUNT__.license.REMOVEATTRIBUTE=[{\"type\": \"LicenseGrant\",\"patchMethod\": \"remove\",\"license\": {\"type\": \"License\",\"code\": \"$(code)$\"}}]","__ACCOUNT__.__PASSWORD__.CREATEOP={\"type\":\"UserPassword\",\"Password\":\"$(__PASSWORD__)$\",\"MustChangePassword\":\"false\"}","__ACCOUNT__.__PASSWORD__.UPDATEOP={\"type\":\"UserPassword\",\"Password\":\"$(__PASSWORD__)$\",\"MustChangePassword\":\"false\"}"` **Note**: The `MustChangePassword` default value will be **False**. If you want to change it, you must change it in the customPayload.
passwordAttribute	This entry holds the name of the target system attribute that is mapped to the __PASSWORD__ attribute of the connector in OIM. **Default value:** Password
statusAttributes	This entry lists the name of the target system attribute that holds the status of an account. For example, for the `__ACCOUNT__` object class that it used for User accounts, the status attribute is disabled. **Default value:**`"__ACCOUNT__.isDisabled"`

Table 3-2 (Cont.) Advanced Settings Parameters

Parameter	Description
statusEnableValue	This entry holds the name attribute for all target values indicating the status as `enabled`.
statusDisableValue	This entry holds the name attribute for all target values indicating the status as `disabled`.

3.3 Attribute Mappings

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

Default Attributes for Eloqua Target Application

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

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

Table 3-3 Default Attributes for Eloqua Target Application

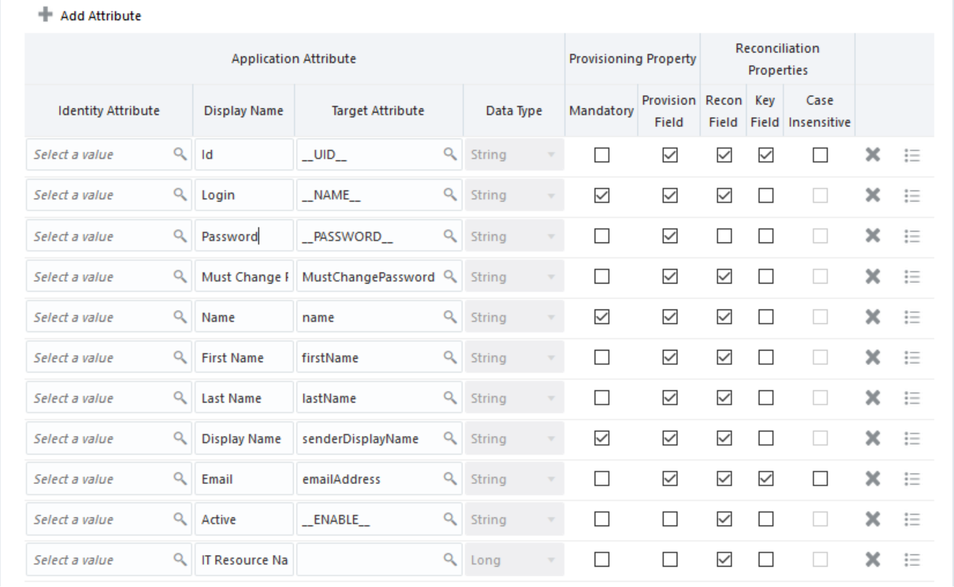
Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Provision Field?	Recon Field?	Key Field?	Case Insensitive?	Advanced Flag Settings
Id	__UID__	String	No	Yes	Yes	Yes	No	Yes
Login	__NAME__	String	Yes	Yes	Yes	Yes	Not applicable	Yes
Name	name	String	Yes	Yes	Yes	Yes	Not applicable	Yes
First Name	firstName	String	No	Yes	Yes	No	Not applicable	Yes
Last Name	lastname	String	No	Yes	Yes	No	Not applicable	Yes
Display Name	senderDisplayName	String	Yes	Yes	Yes	No	Not applicable	Yes
Email	emailAddress	String	No	Yes	Yes	Yes	No	Yes

Table 3-3 (Cont.) Default Attributes for Eloqua Target Application

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Provision Field?	Recon Field?	Key Field?	Case Insensitive?	Advanced Flag Settings
Password	__PASSWORD__	String	No	Yes	No	No	Yes	Yes
Active	__ENABLE__	String	No	No	Yes	No	Not applicable	Yes
IT Resource Name	Not applicable	Long	No	No	Yes	No	Not applicable	Yes

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

Figure 3-1 Default Attribute Mappings for Eloqua User Account



Application Attribute				Provisioning Property	Reconciliation Properties					
Identity Attribute	Display Name	Target Attribute	Data Type	Mandatory	Provision Field	Recon Field	Key Field	Case Insensitive		
Select a value	Id	__UID__	String	☐	☑	☑	☑	☐	✖	☰
Select a value	Login	__NAME__	String	☑	☑	☑	☐	☐	✖	☰
Select a value	Password	__PASSWORD__	String	☐	☑	☐	☐	☐	✖	☰
Select a value	Must Change F	MustChangePassword	String	☐	☑	☑	☐	☐	✖	☰
Select a value	Name	name	String	☑	☑	☑	☐	☐	✖	☰
Select a value	First Name	firstName	String	☐	☑	☑	☐	☐	✖	☰
Select a value	Last Name	lastName	String	☐	☑	☑	☐	☐	✖	☰
Select a value	Display Name	senderDisplayName	String	☑	☑	☑	☐	☐	✖	☰
Select a value	Email	emailAddress	String	☐	☑	☑	☑	☐	✖	☰
Select a value	Active	__ENABLE__	String	☐	☐	☑	☐	☐	✖	☰
Select a value	IT Resource Na		Long	☐	☐	☑	☐	☐	✖	☰

Groups Entitlement

Table 3-4 lists the group forms attribute mappings between the process form fields in Oracle Identity Governance and Eloqua target application attributes. The table lists whether a given attribute is mandatory during provisioning. It also lists whether a given attribute is used during reconciliation and whether it is a matching key field for fetching records during reconciliation.

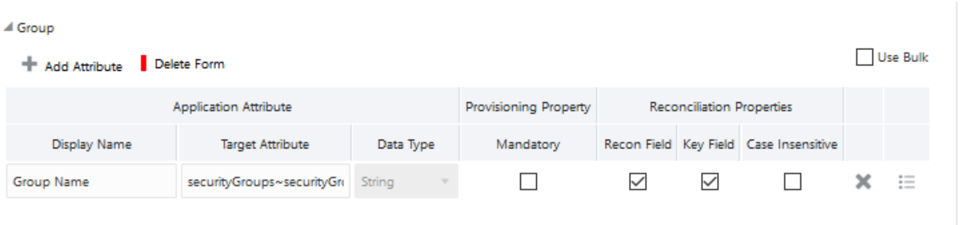
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-4 Default Attribute Mappings for Groups Forms

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Recon Field?	Key Field?	Case Insensitive ?
Group Name	securityGroups~securityGroups	String	No	Yes	Yes	No

Figure 3-2 shows the default attribute groups mapping.

Figure 3-2 Default Attribute Mappings for Groups



Group

Add Attribute | Delete Form | □ Use Bulk

Application Attribute			Provisioning Property	Reconciliation Properties				
Display Name	Target Attribute	Data Type	Mandatory	Recon Field	Key Field	Case Insensitive		
Group Name	securityGroups~securityGr	String	□	☑	☑	□	✖	☰

Licenses Entitlement

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

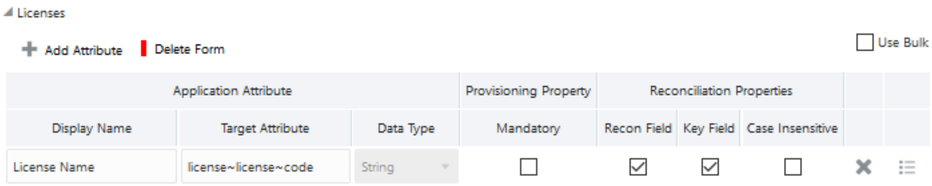
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-5 Default Attribute Mappings for Licenses

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Recon Field?	Key Field?	Case Insensitive ?
License Name	license~license~code	String	No	Yes	Yes	No

Figure 3-3 shows the default attribute licenses mapping.

Figure 3-3 Default Attribute Mappings for Licenses



Licenses

Add Attribute | Delete Form | ☐ Use Bulk

Application Attribute			Provisioning Property	Reconciliation Properties				
Display Name	Target Attribute	Data Type	Mandatory	Recon Field	Key Field	Case Insensitive		
License Name	license~license~code	String	☐	☑	☑	☐	✖	☰

3.4 Correlation Rules, Situations, and Responses for a Target Application

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

Predefined Identity Correlation Rules

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

Table 3-6 lists the default simple correlation rule for Eloqua connector. If required, you can edit the default correlation rule or add new rules. You can create simple correlation rules also. For more information about adding or editing simple or complex correlation rules, see Updating Identity Correlation Rule in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-6 Predefined Identity Correlation Rule for a Eloqua Target Application

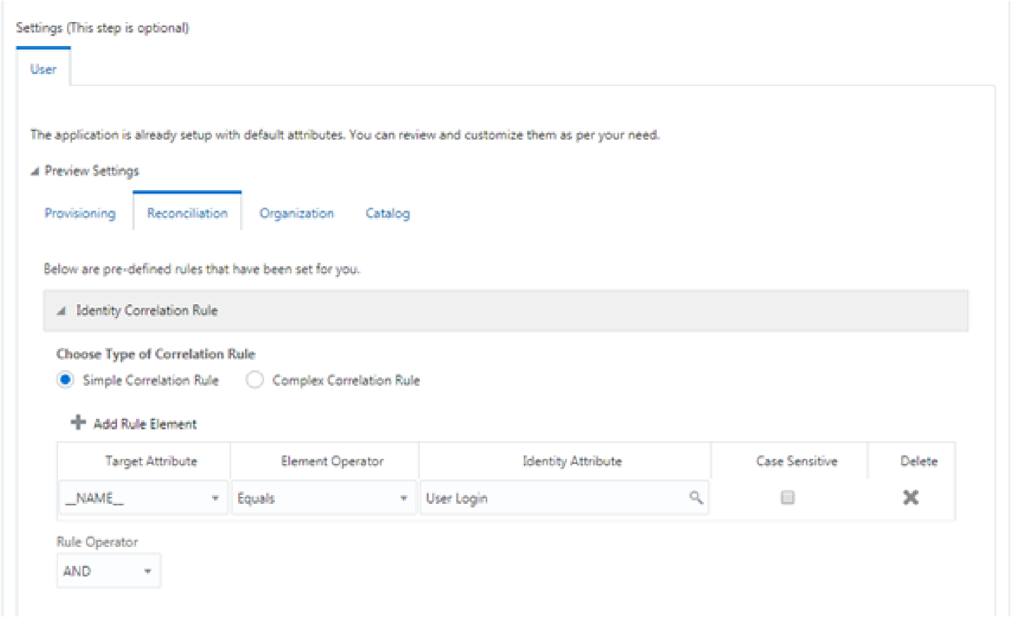
Target Attribute	Element Operator	Identity Attribute	Case Sensitive?	Rule Operator
__NAME__	Equals	User Login	No	AND

In this identity rule:

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

Figure 3-4 shows the simple correlation rule for a Eloqua target application.

Figure 3-4 Simple Correlation Rule for a Eloqua Target Application



Settings (This step is optional)

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

Preview Settings

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

Identity Correlation Rule

Choose Type of Correlation Rule

● Simple Correlation Rule ○ Complex Correlation Rule

Add Rule Element

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

Rule Operator: AND

Predefined Situations and Responses

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

Table 3-7 lists the default situations and responses for Eloqua target application. If required, you can edit these default situations and responses or add new ones. For more information about adding or editing situations and responses, see Creating a Target Application in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-7 Predefined Situations and Responses for a Eloqua Target Application

Situation	Response
No Matches Found	None
One Entity Match Found	Establish Link
One Process Match Found	Establish Link

3.5 Reconciliation Jobs

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

User Reconciliation Jobs

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

Table 3-8 describes the parameters of the Eloqua Full User Reconciliation job.

Table 3-8   Parameters of the Eloqua Full User Reconciliation Job

Parameter	Description
Application name	Name of the AOB application with which the reconciliation job is associated. This value is the same as the value that you provided for the Application Name field while creating your target application. Do *not* change the default value.
Latest Token	This parameter holds the value of the target system attribute that is specified as the value of the Incremental Recon Attribute parameter. The Latest Token parameter is used for internal purposes. By default, this value is empty. **Note**: Do not enter a value for this parameter. The reconciliation engine automatically enters a value in this parameter. **Sample value**: `<String>2017-09-19T14:16:24Z</String>`
Object Type	This parameter holds the name of the object type for the reconciliation run. **Default value**: `User` Do *not* change the default value.
Filter Suffix	Enter the search filter for fetching user records from the target system during a reconciliation run. **Sample value when incremental recon is enabled**: `search=loginName='abcuser'` **Sample value when incremental recon is not enabled**: `search=loginName='abcuser'`
Scheduled Task Name	Name of the scheduled task used for reconciliation. Do *not* modify the value of this parameter.
Incremental Recon Attribute	Enter the name of the attribute that holds the timestamp at which the token record was modified. **Sample value**: `updatedAt`

Delete User Reconciliation Job

The Eloqua Target User Delete Reconciliation job is used to reconcile user data when for target application.

Table 3-9 Parameters of the Eloqua Target User Delete Reconciliation Job

Parameter	Description
Application Name	Name of the application you created for your target system. This value is the same as the value that you provided for the Application Name field while creating your target application. Do *not* modify this value.
Disable User	Enter `yes` if you want the connector to disable accounts (in Oracle Identity Governance) corresponding to accounts deleted on the target system. Enter `no` if you want the connector to revoke accounts in Oracle Identity Governance. Default value: `no`
Object Type	Type of object you want to reconcile. Default value: `User`
Scheduled Task Name	Name of the scheduled task used for reconciliation. Default value: `ELOQUAAOBAPP Eloqua Target User Delete Reconciliation`

Reconciliation Jobs for Entitlements

The following jobs are available for reconciling entitlements:

- Eloqua Group Lookup Reconciliation
- Eloqua Licenses Lookup Reconciliation

The parameters for all the reconciliation jobs are the same.

Table 3-10 Parameters of the Reconciliation Jobs for Entitlements

Parameter	Description
Application Name	Current AOB application name with which the reconciliation job is associated. Default value: `Eloqua` Do *not* modify this value.
Code Key Attribute	Name of the connector attribute that is used to populate the Code Key column of the lookup definition (specified as the value of the Lookup Name attribute). Default value: `__UID__` Do *not* modify this value.

Table 3-10 (Cont.) Parameters of the Reconciliation Jobs for Entitlements

Parameter	Description
Decode Attribute	Name of the connector attribute that is used to populate the Decode column of the lookup definition (specified as the value of the Lookup Name attribute). Default value: `__NAME__`
Lookup Name	Enter the name of the lookup definition in Oracle Identity Governance that must be populated with values fetched from the target system. Depending on the Reconciliation job that you are using, the default values are as follows: • For Eloqua Group Lookup Reconciliation: `Lookup.Eloqua.Group` • For Eloqua Licenses Lookup Reconciliation: `Lookup.Eloqua.License` If you create a copy of any of these lookup definitions, then enter the name of that new lookup definition as the value of the Lookup Name attribute.
Object Type	Enter the type of object you want to reconcile. Depending on the reconciliation job that you are using, the default values are as follows: • For Eloqua Group Lookup Reconciliation: `securityGroups` • For Eloqua Licenses Lookup Reconciliation: `license` **Note**: Do not change the value of this parameter.

4 Installing the Connector in CI Mode

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

- Preinstallation
- Installation
- Postinstallation

4.1 Preinstallation

As a part of preinstallation, ensure all Eloqua users have an Eloqua account with required administrator user access and permissions.

4.2 Installation

You must install the Eloqua connector in Oracle Identity Manager and if required, place the connector code bundle in the Connector Server.

The following topics discuss installing the Eloqua connector:

- Understanding Installation of the Eloqua Connector
- Running the Connector Installer
- Configuring the IT Resource for the Target System

4.2.1 Understanding Installation of the Eloqua Connector

You can run the connector code either locally in Oracle Identity Manager or remotely in a Connector Server.

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

- Run the connector code locally in Oracle Identity Manager. In this scenario, you deploy the connector in Oracle Identity Manager. Deploying the connector in Oracle Identity Manager involves performing the procedures described in Running the Connector Installer and Configuring the IT Resource for the Target System.
- Run the connector code remotely in a Connector Server. In this scenario, you deploy the connector in Oracle Identity Manager, and then, deploy the connector bundle in a Connector Server. See Using an Identity Connector Server in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager* for information about installing, configuring, and running the Connector Server, and then installing the connector in a Connector Server.

4.2.2 Running the Connector Installer

When you run the Connector Installer, it automatically copies the connector files to directories in Oracle Identity Manager, imports connector XML files, and compiles adapters used for provisioning.

> **Note:**
>
> In this guide, the term Connector Installer has been used to refer to the Install Connectors feature of Oracle Identity Manager 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.

   > **Note:**
   >
   > If you are doing it for the first time place the bundle in connector server bundle directory, in that case you need to unzip the bundle before starting the installation.

2. Log in to Oracle Identity System Administration.
3. From the left pane, expand the **Provisioning Configuration** tab and click **Manage Connector**.
4. In the Manage Connector page, click **Install**.
5. From the Connector List, select **Eloqua Connector*RELEASE_NUMBER***.

   This list displays the names and release numbers of connectors whose installation files you copy into the default connector installation directory in Step 1.
   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 **Eloqua Connector *RELEASE_NUMBER.***

6. Click **Load.**
7. 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:

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

8. Click **Exit** to finish the installation procedure.

   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.

   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 Files and Directories in the Eloqua Connector Installation Package .

4.2.3 Configuring the IT Resource for the Target System

The IT resource for the target system is created during connector installation. This IT resource contains connection information about the target system. Oracle Identity Manager uses this information during reconciliation and provisioning.

The Eloqua IT resource is automatically created when you run the Connector Installer. To specify values for the parameters of the IT resource:

1. Log in to Oracle Identity System Administration.
2. In the left pane, under Configuration, click **IT Resource**.
3. In the IT Resource Name field on the Manage IT Resource page, enter `Eloqua` and then click Search.
4. Click the Edit icon 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 4-1 describes each parameter.

Table 4-1 Parameters of the Eloqua IT Resource

Parameter	Description
authenticationType	Enter the type of authentication used by your target system. **Sample value**: `basic`
username	Enter the user name of the target system that you create for performing connector operations. **Sample value**: `IDMsysE10OD01\johnsmith`
password	Enter the password of the target system user account that you create for connector operations. **Sample value**: `password`
host	Enter the host name of the machine hosting your target system. This is a mandatory attribute while creating an application. **Sample value**: `www.example.com`
port	Enter the port number at which the target system is listening. **Sample value**: `443`
proxyHost	Enter the name of the proxy host used to connect to an external target. **Sample value**: `www.example.com`
proxyPassword	Enter the password of the proxy user ID of the target system user account that Oracle Identity Governance uses to connect to the target system.
proxyPort	Enter the proxy port number. **Sample value**: `80`
proxyUser	Enter the proxy user name of the target system user account that Oracle Identity Governance uses to connect to the target system.
sslEnabled	If the target system requires SSL connectivity, then set the value of this parameter to `true`. Otherwise set the value to `false`. **Default value:** `true`

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

4.3 Postinstallation

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

- Configuring Oracle Identity Governance
- Harvesting Entitlements and Sync Catalog
- Managing Logging for the Connector

- Configuring the IT Resource for the Connector Server
- Localizing Field Labels in UI Forms
- Configuring SSL

4.3.1 Configuring Oracle Identity Governance

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

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

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

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

4.3.1.1 Creating and Activating a Sandbox

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

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

4.3.1.2 Creating a New UI Form

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

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

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

4.3.1.3 Publishing a Sandbox

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

1. In Identity System Administration, deactivate the sandbox.
2. Log out of Identity System Administration.

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

4.3.1.4 Updating an Existing Application Instance with a New Form

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

To update an existing application instance with a new form:

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

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

4.3.2 Harvesting Entitlements and Sync Catalog

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

To harvest entitlements and sync catalog:

1. Run the scheduled jobs for lookup field synchronization listed in Reconciliation Jobs.
2. Run the Entitlement List scheduled job to populate Entitlement Assignment schema from child process form table.
3. Run the Catalog Synchronization Job scheduled job.

See Also:

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

4.3.3 Managing Logging for the Connector

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

The following topics provide detailed information about logging:

- Understanding Log Levels
- Enabling Logging

4.3.3.1 Understanding Log Levels

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

- SEVERE.intValue()+100

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

  This level enables logging of information about errors that might allow Oracle Identity 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 4-2.

Table 4-2 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.

4.3.3.2 Enabling Logging

Perform this procedure to enable logging in Oracle WebLogic Server.

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='Eloqua-handler'
level='[LOG_LEVEL]'class='oracle.core.ojdl.logging.ODLHandlerFact
ory'>
    <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.GENERICREST"
level="[LOG_LEVEL]" useParentHandlers="false">
    <handler name="Eloqua-handler"/>
    <handler name="console-handler"/>
</logger>


<logger name="ORG.IDENTITYCONNECTORS.RESTCOMMON"
level="[LOG_LEVEL]" useParentHandlers="false">
    <handler name="Eloqua-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 4-2 lists the supported message type and level combinations. Similarly, replace [FILE_NAME] with the full path and name of the log file in which you want log messages to be recorded. The following blocks show sample values for **[LOG_LEVEL]** and **[FILE_NAME]:**

```
<log_handler name='Eloqua-handler'
level='NOTIFICATION:1'class='oracle.core.ojdl.logging.ODLHandlerF
actory'>
    <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.GENERICREST"
level="NOTIFICATION:1" useParentHandlers="false">
    <handler name="Eloqua-handler"/>
    <handler name="console-handler"/>
</logger>

<logger name="ORG.IDENTITYCONNECTORS.RESTCOMMON"
level="NOTIFICATION:1" useParentHandlers="false">
    <handler name="Eloqua-handler"/>
    <handler name="console-handler"/>
</logger>
```

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

2. Save and close the file.
3. 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.

4.3.4 Configuring the IT Resource for the Connector Server

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

After you create the application for your target system, the connector creates a default IT resource for the Connector Server. The name of this default IT resource is `Eloqua Connector Server`.

In Oracle Identity System Administration, search for and edit the Eloqua Connector Server IT resource to specify values for the parameters of IT resource for the Connector Server listed in Table 4-2. For more information about searching for IT resources and updating its parameters, see Managing IT Resources in *Oracle Fusion Middleware Administering Oracle Identity Governance*.

Table 4-3 Parameters of the IT Resource for the Eloqua Connector Server

Parameter	Description
Host	Enter the host name or IP address of the computer hosting the Connector Server. Sample value: `HostName`
Key	Enter the key for the Connector Server.
Port	Enter the number of the port at which the Connector Server is listening. Sample value: `8763`
Timeout	Enter an integer value which specifies the number of milliseconds after which the connection between the Connector Server and Oracle Identity Governance times out. If the value is zero or if no value is specified, the timeout is unlimited. Sample value: `0` (recommended value)
UseSSL	Enter `true` to specify that you will configure SSL between Oracle Identity Governance and the Connector Server. Otherwise, enter `false.` Default value: `false` **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*.

4.3.5 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. Resource bundles are available in the connector installation media.

To localize field labels that is added to the 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 the following file in a text editor:

   ```
   SAVED_LOCATION\xliffBundles\oracle\iam\ui\runtime\BizEditorBundle.xl
   f
   ```

   > **Note:**
   >
   > You will not be able to view the BizEditorBundle.xlf file unless you complete creating the application for your target system or perform any customization such as creating a UDF.

6. Edit the BizEditorBundle.xlf file in the following manner:

   a. Search for the following text:

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

   b. Replace with the following text:

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

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

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

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

   ```
   <trans-unit id="$
   {adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResou
   rceBundle']
   ['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity
   .userEO.UD_Name_c_description']}">
   <source>Name</source><target/>
   </trans-unit>
   id="sessiondef.oracle.iam.ui.runtime.form.model.RSAForm.entity.El
   oquaFormEO.UD_NAME __c_LABEL"><source>Name</source><target/>"
   </trans-unit>
   ```

d. Open the resource file from the connector package, for example Eloqua_ja.properties, and get the value of the attribute from the file, for example,

```
global.udf.UD_USR_EQ_LOGIN =\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.BaseRuntimeResou
rceBu ndle']
['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity
.use rEO.global.udf.UD_USR_EQ_LOGIN __c_description']}">
<source>Login</source>
<target>u30A2\u30AB\u30A6\u30F3\u30C8\u540D</target>
</trans-unit> <trans-
unitid="sessiondef.oracle.iam.ui.runtime.form.model.Eloqua.entity
 sEO.global.udf.UD_USR_EQ_LOGIN__c_LABEL">
<source>Login</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.

4.3.6 Configuring SSL

Configure SSL to secure data communication between Oracle Identity Governance and the Eloqua target system.

> **Note:**
>
> If you are using this connector along with a Connector Server, then there is no need to configure SSL. You can skip this section.

To configure SSL:

1. Obtain the SSL public key certificate of Eloqua.

2. Copy the public key certificate of Eloqua to the computer hosting Oracle Identity Governance.
3. Run the following `keytool` command to import the public key certificate into the identity key store in Oracle Identity Governance:

   ```
   keytool -import -alias ALIAS -trustcacerts -file CERT_FILE_NAME -
   keystore KEYSTORE_NAME -storepass PASSWORD
   ```

   In this command:

   - *ALIAS* is the public key certificate alias.
   - *CERT_FILE_NAME* is the full path and name of the certificate store (the default is cacerts).
   - *KEYSTORE_NAME* is the name of the keystore.
   - *PASSWORD* is the password of the keystore.

   The following is a sample value for this command:

   ```
   keytool -import -alias serverwl -trustcacerts -file supportcert.pem -
   keystore client_store.jks -storepass weblogic1
   ```

   > **Note:**
   >
   > - Change the parameter values passed to the `keytool` command according to your requirements. Ensure that there is no line break in the keytool arguments
   > - Ensure that the system date for Oracle Identity Governance is in sync with the validity date of the SSL certificate to avoid any errors during SSL communication.

5
Using the Connector

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

- Configuring Reconciliation
- Configuring Reconciliation Jobs
- Configuring Provisioning
- Cloning the Connector
- Defining the Connector
- Uninstalling the Connector

5.1 Configuring Reconciliation

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

This section discusses the following topics related to configuring reconciliation:

- Performing Full and Incremental Reconciliation
- Performing Limited Reconciliation

5.1.1 Performing Full and Incremental Reconciliation

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

For a Target application, you can only 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.

To perform an incremental reconciliation run, set the value of the Incremental Recon Attribute to `updatedAt`, and then run the Eloqua User Target Reconciliation job. At the end of the reconciliation run, the Latest Token parameter of the reconciliation job for the user record reconciliation is automatically updated. From the next reconciliation run onward, only records created after this time stamp are considered for reconciliation. This is incremental reconciliation.

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

See Reconciliation Jobs for information about this reconciliation jobs.

5.1.2 Performing Limited Reconciliation

Limited or **filtered** reconciliation is the process of limiting the number of records being reconciled based on a set filter criteria.

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

All users are associated with a unique system ID, also known as `loginName`. The `loginName` attribute is present in the target system and OIG. Filtered reconciliation is performed using the `loginName` as a filter suffix attribute.

You can perform limited reconciliation by creating filters for the reconciliation module. This connector provides a Filter Suffix attribute (a scheduled task attribute) that allows you to use the `loginName` attribute of the target system to filter target system records. The `loginName` is appended to the endpoint URL. When this endpoint URL is reconciled, all record reconciliation is limited to this filter suffix attribute. A sample filter suffix value is `search=loginName=raisee`. The value provided in the filter suffix parameter varies in accordance with the target system.

While creating the application, follow the instructions in Configuring Reconciliation to specify attribute values.

5.2 Configuring Reconciliation Jobs

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

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

To configure a reconciliation job:

1. Log in to Identity System Administration.
2. In the left pane, under System Management, click **Scheduler**.
3. Search for and open the scheduled job as follows:
   a. In the Search field, enter the name of the scheduled job as the search criterion. Alternatively, you can click **Advanced Search** and specify the search criterion.
   b. In the search results table on the left pane, click the scheduled job in the Job Name column.
4. On the Job Details tab, you can modify the parameters of the scheduled task:
   - **Retries**: Enter an integer value in this field. This number represents the number of times the scheduler tries to start the job before assigning the Stopped status to the job.
   - **Schedule Type**: Depending on the frequency at which you want the job to run, select the appropriate schedule type. See Creating Jobs in *Oracle Fusion Middleware Administering Oracle Identity Governance*.

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

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

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

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

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

5.3 Configuring Provisioning

Learn about performing provisioning operations in Oracle Identity Governance and the guidelines that you must apply while performing these operations.

- Guidelines on Performing Provisioning Operations
- Performing Provisioning Operations

5.3.1 Guidelines on Performing Provisioning Operations

This guideline provides information on what to do while performing provisioning operations.

For a Create User provisioning operation, you must specify a value for the Name, Login, and DisplayName fields. These are mandatory fields.

5.3.2 Performing Provisioning Operations

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

To perform provisioning operations in Oracle Identity Governance:

1. Log in to Identity Self Service.
2. Create a user as follows:
   a. In Identity Self Service, click **Manage**. The Home tab displays the different Manage option. Click **Users**. The Manage Users page is displayed.
   b. From the Actions menu, select **Create**. Alternatively, you can click **Create** on the toolbar. The Create User page is displayed with input fields for user profile attributes.
   c. Enter details of the user in the Create User page.

3. On the Account tab, click **Request Accounts**.
4. In the Catalog page, search for and add to cart the application instance for the connector that you configured earlier, and then click **Checkout**.
5. Specify value for fields in the application form and then click **Ready to Submit**.
6. Click **Submit**.

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

5.4 Cloning the Connector

You can clone the Eloqua connector by setting new names for some of the objects that comprise the connector.

The outcome of the process is a new connector XML file. Most of the connector objects, such as Resource Object, Process Definition, Process Form, IT ResourceType Definition, IT Resource Instances, Lookup Definitions, Adapters, Reconciliation Rules and so on in the new connector XML file have new names.

> **See Also:**
>
> Managing Connector Lifecycle of *Oracle Fusion Middleware Administering Oracle Identity Governance* for detailed information about cloning connectors and the steps mentioned in this section.

After a copy of the connector is created by setting new names for connector objects, some objects might contain the details of the old connector objects. Therefore, you must modify the following Oracle Identity Manager objects to replace the base connector artifacts or attribute references with the corresponding cloned artifacts or attributes:

- **IT Resource**: The cloned connector has its own set of IT resources. You must configure both the cloned IT resources, Eloqua and Connector Server, and provide the reference of the cloned Connector Server IT Resource in the cloned Eloqua IT resource. Ensure you use the configuration lookup definition of the cloned connector.
- **Scheduled Task**: The values of the Resource Object Name and IT Resource scheduled taskattributes in the cloned connector refer to the values of the base connector.Therefore, these values (values of the Resource Object Name and IT resource scheduled task attributes that refer to the base connector) must be replaced withthe new cloned connector artifacts.
- **Lookup Definition**: Verify the lookup entries in all lookup definitions to ensure that there are no references of old process forms. If there are any, then change it to the corresponding new form. For example, after cloning, the Lookup.Eloqua.UM.ProvAttrMap lookup definition contains a reference to a child

table such as `UD_ELOQUA`GroupName[LOOKUP]`. You must change this to include the new value, for example, `UD_ELOQUA2`GroupName[LOOKUP]`.

- **Process Tasks**: After cloning, you notice that all event handlers attached to the process tasks are the cloned ones. Therefore, no changes are required for process tasks in parent forms. This is because the adapter mappings for all process tasks related to parent forms are updated with cloned artifacts.
- **Localization Properties**: You must update the resource bundle of a user locale with new names of theprocess form attributes for proper translations after cloning the connector. You can modify the properties file of your locale in the resources directory of the connector bundle. For example, the process form attributes are referenced in the Japanes eproperties file, EloquaIdC_ja.properties, as `global.udf.UD_ELOQUA_FULLNAME`. During cloning, if you change the process form name from `UD_ELOQUA` to `UD_ELOQUA1`, then you must update the process form attributes to `global.udf.UD_ELOQUA1_FULLNAME`.

5.5 Defining the Connector

By using the Administrative and User Console, you can define a customized or reconfigured connector. Defining a connector is equivalent to registering the connector with Oracle Identity Manager.

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 about the procedure to define connectors.

5.6 Uninstalling the Connector

Uninstalling the connector deletes all the account related data associated with resource objects of the connector. You use the Uninstall Connectors utility to uninstall a connector.

For detailed instructions on uninstalling the connector for any reason, see Uninstalling Connectors in *Oracle Fusion Middleware Administering Oracle Identity Manager.*

6

Extending the Functionality of the Connector

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

This section discusses the following topics:

- Configuring Transformation and Validation of Data
- Configuring Action Scripts
- Configuring the Connector for Multiple Installations of the Target System

6.1 Configuring Transformation and Validation of Data

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

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

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

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

6.2 Configuring Action Scripts

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

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

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

6.3 Configuring the Connector for Multiple Installations of the Target System

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

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

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

7
Known Issues and Limitations

These are the known issues and limitations associated with the Eloqua connector.

Default Security Group "Everyone" Added on Eloqua Application

A security group named "Everyone" is added to every user created on the Eloqua application by default. Hence, when a user runs the reconciliation job, this security group is also automatically reconciled.

Workaround: As a workaround, users can perform provisioning operations with the security group "Everyone" assigned.

Note: You should not remove all groups assigned to a user. If all groups are removed, delete reconciliation operation will be impacted.

Dependent Licenses Automatically Assigned/Unassigned to Security Groups

Eloqua application automatically assigns inherited licenses to cetain security groups. Therefore, when a user is assigned any such security groups, these inherited licenses are also automatically assigned to the user after performing a reconciliation operation. These licenses are reflected in the Eloqua application and Oracle Identity Governance.

However, when any of these secuity groups are assigned to a user on Oracle Identity Governance, these groups alone are assigned. After a reconciliation operation is performed, the inherited licenses assigned to these groups are automatically asigned to the user. Similarly, when any of these secuity groups are unassigned for a user on Oracle Identity Governance, these groups alone are unassigned. Only after a reconciliation operation is performed, the inherited licenses assigned to these groups are automatically unassigned from the user.

Workaround: There is no workaround for this issue as this is an expected behavior of the Eloqua application.

A

Files and Directories in the Eloqua Connector Installation Package

These are the components of the connector installation package that comprise the Eloqua connector.

Table A-1 Files and Directories in the Eloqua Connector Installation Package

File in the Installation Package	Description
bundle/ org.identityconnectors.genericrest-12.3.0.jar	This JAR is the ICF connector bundle. **Note**: If you try to install any other generic rest connector like Office 365 on top of the Eloqua connector, then the generic rest bundle jar `org.identityconnectors.genericrest-12.3.0.jar` must be replaced with the latest generic rest jar or replaced with the Eloqua bundle jar.
configuration/Eloqua-CI.xml	This file is used for installing a CI-based connector. This XML file contains configuration information that is used by the Connector Installer during connector installation.
Files in the resources directory	Each of these resource bundles contains language-specific information that is used by the connector. During connector installation, these resource bundles are copied to Oracle Identity database. **Note:** A **resource bundle** is a file containing localized versions of the text strings that are displayed on the Administrative and User Console. These text strings include GUI element labels and messages.
xml/Eloqua-target-template.xml	This file contains definitions for the connector objects required for creating a Target application. It includes certain details required to connect Oracle Identity Governance with the target system. It also includes configuration details specific to your target system, attribute mappings, correlation rules, and reconciliation jobs.
lib/eloqua-enable-transformation.jar	This file is used to support enable and disable functionality in the CI mode of connector installation.
xml/Eloqua-ConnectorConfig.xml	This XML file contains definitions for the connector components. These components include the following: • IT resource type • Process form • Process task and adapters (along with their mappings) • Resource object • Provisioning process • Prepopulate rules • Lookup definitions • Scheduled tasks