Oracle® Identity Governance

Configuring SAP Ariba Connector

12c (12.2.1.3.0)
F87665-02

Oracle Identity Governance Configuring SAP Ariba Connector, 12c (12.2.1.3.0)

F87665-02

Primary Authors: Maya, Syam Battu

Contributing Authors: Brunda M

Contents

1 Introduction to the Connector

2 Creating an Application by Using the Connector

List of Figures

List of Tables

Preface

This guide describes the connector that is used to onboard the SAP Ariba application to Oracle Identity Governance.

Audience

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

Related Documents

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

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

For information about Oracle Identity Manager Connectors 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.

1
Introduction to the Connector

This chapter introduces the SAP Ariba Application 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 SAP Ariba Connector lets you create and onboard SAP Ariba 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**.

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 sections provide a high-level overview of the 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 SAP Ariba connector.

Table 1-1 Certified Components

Component	Requirement for AOB Application
Oracle Identity Governance or Oracle Identity Manager	You can use any one of the following releases: • Oracle Identity Governance release 12c PS4 (12.2.1.4.0) or later. • Oracle Identity Governance release 12c PS3 (12.2.1.3.0) or later.
Oracle Identity Governance or Oracle Identity Manager JDK	JDK 1.8 and later
Target systems	SAP Ariba
Connector Server	11.1.2.1.0 or 12.2.1.3.0
Connector Server JDK	JDK 1.8 and later
Target API version	SAP Ariba v1

1.2 Usage Recommendation

If you are using Oracle Identity Governance 12*c* (12.2.1.3.0) or later, 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
- 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

This is 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
Enable user	Yes
Disable user	Yes
Group Grant Management	
Assign and Revoke Group	Yes

1.5 Connector Architecture

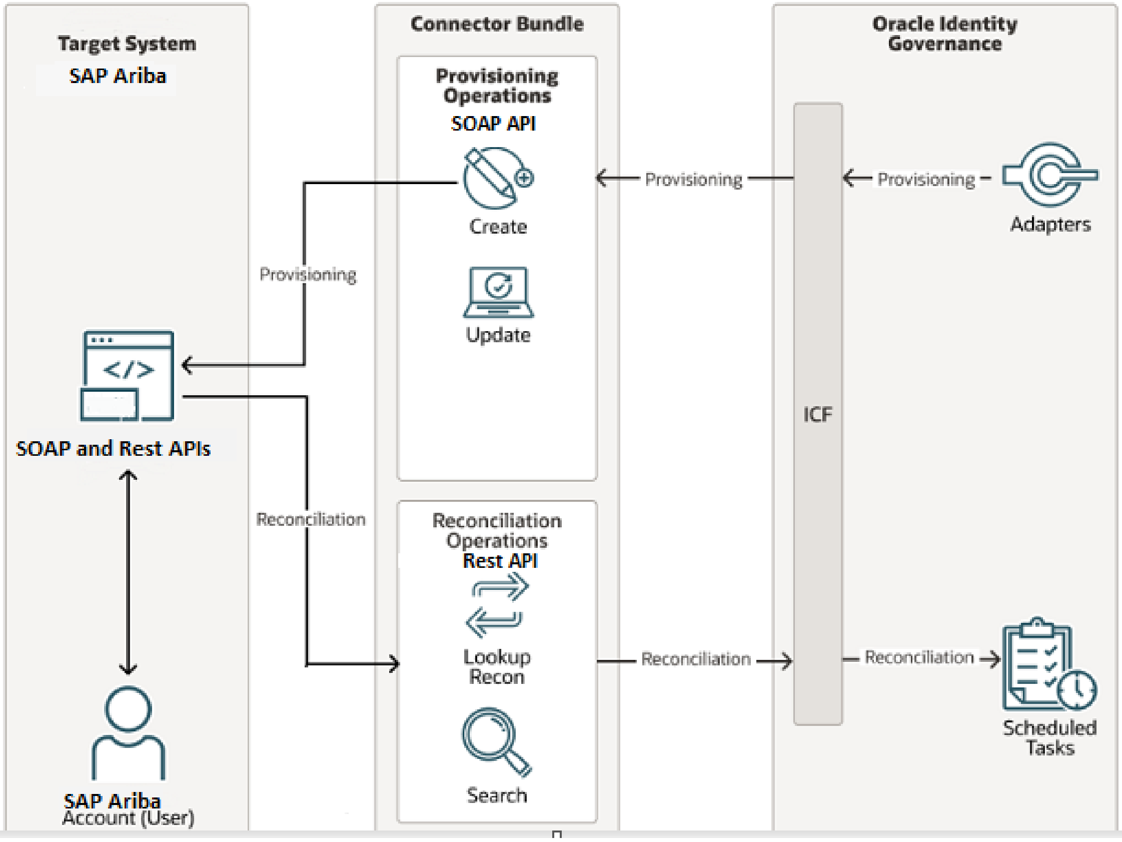
The SAP Ariba is implemented by using the Identity Connector Framework (ICF).

The ICF is a component that is required 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.

The following figure shows the architecture of the SAP Ariba.

The connector is configured to run in one of the following modes:

Figure 1-1 SAP Ariba Connector Architecture



- **Account management**
  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 and updating users on the target system through Oracle Identity Governance. During provisioning, the Adapters invoke ICF operation, ICF in turn invokes create operation on the SAP Ariba Identity Connector Bundle and then the bundle calls the target system API (SAP Ariba SOAP API) for provisioning operations. The SOAP 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.
  - **Target resource reconciliation**
    During reconciliation, a scheduled task invokes an ICF operation. ICF in turn invokes a search operation on the SAP Ariba Identity Connector Bundle and then the bundle calls SAP Ariba REST API for Reconciliation operation. The REST 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 Governance.

    Each record fetched from the target system is compared with SAP Ariba resources that are already provisioned to OIG Users. If a match is found, then the update made to the SAP Ariba record in OIG from the target system. If no match is found, then the Name of the record is compared with the User Login of each OIG User, if a match is found, then data in the target system record is used to provision an SAP Ariba resource to the OIG User.

The SAP Ariba Identity Connector Bundle communicates with the SAP Ariba SOAP and REST API using the HTTPS protocol. The SAP Ariba API provides programmatic access to SAP Ariba through SOAP and REST API endpoints. Application can use the SOAP and REST API to perform create, read, and update operations for users, assigning and removal of groups.

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

- **SAP Ariba User Management**:
  An organization using SAP Ariba 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 easy way is to install the SAP Ariba 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 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

1.7 Connector Features

The features of the connector include support for connector server, User Provisioning, full reconciliation, Incremental reconciliation and limited (Filtered) reconciliation.

The below table provides the list of features supported by the AOB application.

Table 1-3 Supported Connector Features Matrix

Feature	AOB Application
User Provisioning	Yes
Full reconciliation	Yes
Incremental reconciliation	Yes

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

Feature	AOB Application
Limited (Filtered) reconciliation	Yes
Use connector server	Yes
Transformation and validation of account data	Yes
Perform connector operations in multiple domains	Yes
Support for paging	Yes
Test connection	Yes
SAP Ariba Group assignment	Yes

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

- User Provisioning
- Full Reconciliation and Incremental Reconciliation
- Limited (Filtered) Reconciliation
- Support for the Connector Server
- Transformation and Validation of Account Data
- Support for Cloning Applications and Creating Instance Applications
- Secure Communication to the Target System

1.7.1 User Provisioning

User provisioning involves creating or modifying the account data on the target system through Oracle Identity Governance.

> **Note:**
>
> For more information, see Performing Provisioning Operations

1.7.2 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 timestamp stored in the Sync Token attribute of the scheduled job.

For more information, see Performing Full and Incremental Reconciliation.

1.7.3 Limited (Filtered) 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 (Filtered) Reconciliation.

1.7.4 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.

1.7.5 Transformation and Validation of Account Data

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

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

1.7.6 Support for Cloning Applications and Creating Instance Applications

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

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

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

1.7.7 Secure Communication to the Target System

To provide secure communication to the target system, SSL is required. You can configure SSL between Oracle Identity Governance and the Connector Server and between the Connector Server and the target system.

If you do not configure SSL, passwords can be transmitted over the network in clear text. For example, this problem can occur when you are creating a user or modifying a user's password.

For more information, see Configuring SSL.

2 Creating an Application by Using the Connector

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

- Prerequisites for Creating an Application By Using the Connector
- Process Flow for Creating an Application By Using the Connector
- Creating an Application By Using the SAP Ariba Cloud Connector

2.1 Prerequisites for Creating an Application By Using the Connector

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

- Configuring Target System to Perform Connector Operations
- Get the Authentication Server URL
- Partition and Variant values can be fetched from the WSDL
- Configure SOAP End point URL in relUrl's for connector operation in the Advance Configuration
- Configure REST End point URL in relUrl's for connector operation in the Advance Configuration
- More information for Basic and Advance configurations while creating the application
- Downloading the Connector Installation Package

2.1.1 Configuring Target System to Perform Connector Operations

Prerequisites:

You must be a member of the Customer Administrator or Integration Admin group, or a group with the Administrator or Integration Admin role.

Context:

An end point consists of the URL and authentication information that controls access to the end point. There are two types of end points: inbound and outbound. Inbound end points are located within the Ariba service. Outbound end points are located in an external service.

- Configuring Target System for Provisioning Operation
- Configuring Target System for Reconciliation Operation

2.1.1.1 Configuring Target System for Provisioning Operation

Use this procedure to configure an end point for SOAP web services.

1. Log in to SAP Ariba instance with admin credentials.

2. On the Ariba Administrator dashboard, click **Manage** and select **Administration** from the drop-down.
3. Expand the **Integration Manager** option and select **End Point Configuration**.
4. To create a new End Point:
   a. Click **Create New**.
   The page **End Point Configuration - Create End Point** opens.
5. In the **Name** field, enter a name for the end point and select the type as Inbound of end point.
6. Navigate to the **HTTP Authentication** section, to use HTTP Basic Authentication.
   a. Enter the user ID in the **Login** field and the password in the **Password** and **Verify Password** fields.
7. Click **Save**.

2.1.1.2 Configuring Target System for Reconciliation Operation

This procedure allows you to configure an end point for the REST web services.

> **Note:**
>
> Refer to KB0399724 in SAP Ariba connect Portal.

1. Login to the SAP Ariba Developer Portal.
2. Click **Manage** from the left navigation menu.
3. Click **Applications**.
4. Search for the application you want enabled.
5. Click **Actions** > **Request API Access**.
6. Complete the following:
   - Select the API name in the **API Names** drop-down. This is the API that you want to access using this application.
   - Enter the realm name in the **Realm Name** field, which is the site that you want the application to be enabled for.
   - Enter the ANID in the **AN-ID** field.
   - Select the following options corresponding to the realm type:
     - **Production**
       OR
     - **Test**
   - Click **Submit**.

   > **Note:**
   >
   > Approve Process to be taken by a user with **Organization Admin** privileges:

7. Once the application is approved, admin needs to click on **Generate OAuth Secret** under **Actions**.
8. Capture the details for **Application Key, OAuth Client Id, OAuth Secret, Base64 Encoded Client and Secret** on the screen after generation.

2.1.2 Get the Authentication Server URL

1. Login to the SAP Ariba Developer Portal.
2. Go to **Discover** section, and navigate to **STRATEGIC SOURCING**.
3. Search for **Master Data Retrieval API for Sourcing**.
4. Copy the **OAuth Server URL** and append **v2**.
   Example: https://< OAuth Server URL >/v2

2.1.3 Partition and Variant values can be fetched from the WSDL

1. Login to Ariba instance console with admin credentials.
2. In the Ariba Administrator dashboard, click **Manage** and select **Administration** from drop-down.
3. Expand **Integration Manager** and select **Integration Configuration**.
4. Search for **Import Users** Web Service, Task name=**Import User**.
   Next and click and open the **Import Users** Web Service.
5. Click **View WSDL** and search for **vrealm** in wsdl.

   > **Note:**
   >
   > Refer to **KB0400499** in SAP Ariba connect Portal.

6. If you find vrealm_1234 in WSDL, you can use vrealm_1234 as Variant and prealm_1234 as Partition.

2.1.4 Configure SOAP End point URL in relUrl's for Connector Operation in the Advance Configuration

1. Login to Ariba instance console with admin credentials.
2. In the Ariba Administrator dashboard, click **Manage** and select **Administration** from the drop-down.
3. Expand **Integration Manager** and select **Integration Configuration**.
4. Search for **Import Users** Web Service, Task name=**Import User**
   Next and click and open the **Import Users** Web Service.
5. Copy the URL value, its required for Connector application Advanced Configuration relURIs attribute **CreateUserURL**.
   Example: https://<hostname.com>/sourcing/Soap/<realmname>/UserImport

   .
6. Repeat above step 2 and 3 and search for **Add Users to Group** Web Service, Task name= **Add Users to Group**.
   Click and open the **Add Users to Group** Web Service.

7. Copy the URL value, its required for Connector application Advanced Configuration relURIs attribute **GroupsAssignURL**.
   Example: https://<hostname.com>/sourcing/Soap/<realmname>/AddUsersToGroup
8. Repeat above step 2 and 3 and Search for **Remove User from Group** Web Service, Task name= **Remove User from Group**
   Next and click and open the **Remove User from Group** Web Service.
9. Copy the URL value, its required for Connector application Advanced Configuration relURIs attribute **GroupsRemoveURL**.
   Example : https://<hostname.com>/sourcing/Soap/<realmname>/RemoveUsersFromGroup

2.1.5 Configure REST End point URL in relUrl's for connector operation in the Advance Configuration

1. Login to the SAP Ariba Developer Portal.
2. In the **Discover** section, navigate to **STRATEGIC SOURCING**.
3. Search for **Master Data Retrieval API for Sourcing**
4. Copy the URL value from the **Production and Test URL**.
5. Append **/entities/users?$includeInactive=true** after the Production URL.
6. Replace this final URL with **$(UserReconURL)$** in the relURL.
   Example: **$(UserReconURL)$** =https://**<hostname.com>**/api/sourcing-mds-search/v1/prod/entities/users?$includeInactive=true
7. Append **/entities/groups?** after Production URL and replace the final URL with **$(GroupLookupURL) $**
   Example: **$(GroupLookupURL) $**=https://**<hostname.com>**/api/sourcing-mds-search/v1/prod/entities/groups?
8. Similarly replace **$(OrganizationLookupURL)$**,**$(CurrencyLookupURL)$** and **$(LocaleIdLookupURL)$** with **/entities/organizations?** , **/entities/currency?** and **/entities/localeids?** respectively.
   Examples:

   $(OrganizationLookupURL)$= https://<hostname.com>/api/sourcing-mds-search/v1/prod/entities/organizations?&$filter=SystemID%20eq%20'[Buyer]'

   $(CurrencyLookupURL)$= https://openapi.ariba.com/api/sourcing-mds-search/v1/prod/entities/currency?

   $(LocaleIdLookupURL)$ = https://openapi.ariba.com/api/sourcing-mds-search/v1/prod/entities/localeids?

2.1.6 More information for Basic and Advance configurations while creating the application

- authenticationServerUrl: < Refer Section 2.1.2>
- username and password: Refer Section 2.1.1.1 step 6.
- ClientID: Refer 2.1.1.2 section, steps 7 and 8 take the value of OAuth Client Id
- clientSecret: Refer 2.1.1.2 section, steps 7 and 8 take the value of OAuth Secret
- apiKey: Refer 2.1.1.2 section, steps 7 and 8 take the value of Application Key

- In the SAP Ariba Management URL, you will find the realm name which can be used as a XRealm value
- Partition and Variant, Refer Section 2.1.3 step 5 and 6.
  relURIs:

  Default Value:

  ```
  "__ACCOUNT__.CREATEOP=$(CreateUserURL)$","__ACCOUNT__.UPDATEOP=$
  (UpdateUserURL)$","__ACCOUNT__.SEARCHOP=$
  (UserReconURL)$","__ACCOUNT__.__GROUP__.GROUPLKP=$
  (GroupLookupURL)$","__ACCOUNT__.ORGANIZATION.Lookup=$
  (OrganizationLookupURL)$","__ACCOUNT__.CURRENCY.Lookup=$
  (CurrencyLookupURL)$","__ACCOUNT__.LOCALEID.Lookup=$
  (LocaleIdLookupURL)$","__ACCOUNT__.__GROUP__.ADDATTRIBUTE=$
  (GroupsAssignURL)$","__ACCOUNT__.__GROUP__.REMOVEATTRIBUTE=$
  (GroupsRemoveURL)$"
  ```

  In CREATEOP , `$(CreateUserURL)$` replace with **Import Users Web Service URL** : Refer 2.1.2 Steps4

  In UPDATEOP, `$(UpdateUserURL)$` replace with **Import Users Web Service URL**: Refer 2.1.2 Steps 4

  In SEARCHOP

  `$(UserReconURL)$`, Refer section 2.1.3, step 5.

  `$(GroupLookupURL)$`, Refer section 2.1.4, step8.

  `$(OrganizationLookupURL)$`, Refer section 2.1.4, step9.

  `$(CurrencyLookupURL)$`, Refer section 2.1.4, step9.

  `$(LocaleIdLookupURL)$`, Refer section 2.1.4, step9.

  `$(GroupsAssignURL)$` replace with Add **Users to Group Web Service URL:** Refer 2.1.4 Step7

  `$(GroupsRemoveURL)$"` replace with Remove **User from Group URL:** Refer 2.1.4 Step 9

2.1.7 Downloading the Connector Installation Package

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

To download the connector installation package:

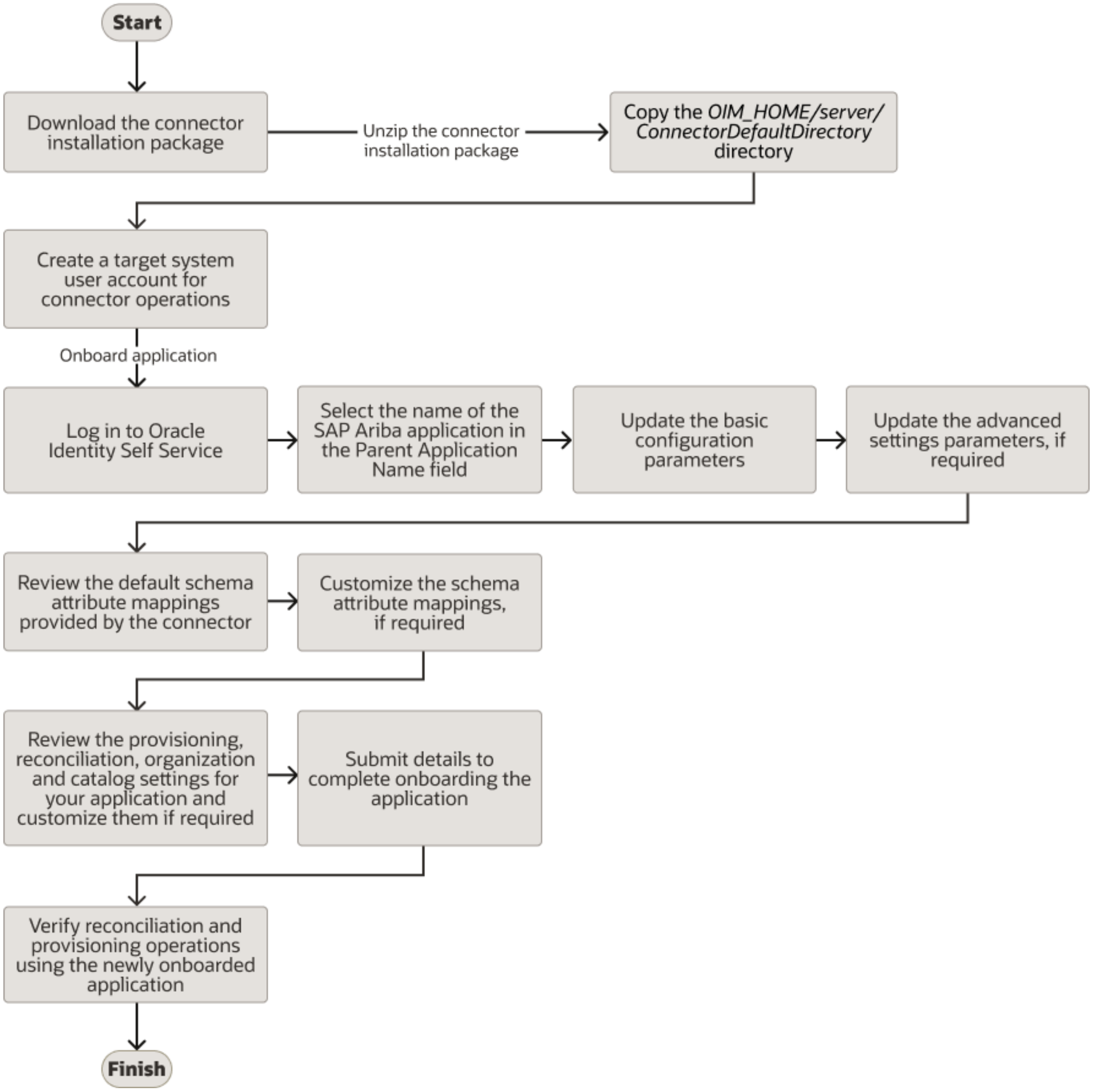
1. Navigate to the OTN website at http://www.oracle.com/technetwork/middleware/id-mgmt/downloads/connectors-101674.html.
2. Click **OTN License Agreement** and read the license agreement.
3. Select the **Accept License Agreement** option.
   You must accept the license agreement before you can download the installation package.
4. Download and save the installation package to any directory on the computer hosting Oracle Identity Governance.
5. Extract the contents of the installation package to any directory on the computer hosting Oracle Identity Governance. This creates a directory named *CONNECTOR_NAME-RELEASE_NUMBER*.
6. Copy the *CONNECTOR_NAME-RELEASE_NUMBER* directory to the *OIG_HOME*/server/ConnectorDefaultDirectory directory.

2.2 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.

The following 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 the *OIM_HOME/server/ConnectorDefaultDirectory* directory
Create a target system user account for connector operations
Onboard application
Log in to Oracle Identity Self Service
Select the name of the SAP Ariba application in the Parent Application Name field
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
Finish

2.3 Creating an Application By Using the SAP Ariba Cloud 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 regarding each step 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.

> **Note:**
>
> - **Configuring the Connector** of 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

Configure connection-related parameters while creating a target application. These parameter values will be used to connect Oracle Identity Governance with your target system and perform connector operations. In addition, you can view and edit attribute mappings between the process form fields in Oracle Identity Governance and target system columns, predefined correlation rules, situations and responses, and reconciliation jobs.

- Basic Configuration Parameters
- Advanced Settings Parameters
- Attribute Mappings
- Correlation Rules
- Reconciliation Jobs

3.1 Basic Configuration Parameters

These are the connection-related parameters that Oracle Identity Governance requires to connect to SAP Ariba Application.

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

Table 3-1 Parameters in the Basic Configuration

Parameter	Mandatory?	Description
grantType	Yes	The Client Credentials grant type is used by clients to obtain an access token. **Default value**: client_credentials
authenticationServerUrl	Yes	Enter the URL of the authentication server that validates the client ID and client secret for your target system. **Default value**: null
clientID	No	Enter the client identifier (a unique string) issued by the authorization server to your client application during the registration process. This client ID is obtained while performing the procedure described in Configuring the Newly Added Application. **Default value**: null
clientSecret	No	Enter the secret key used to authenticate the identity of your client application. You obtained the secret key while performing the procedure described in Configuring the Newly Added Application. **Default value**: null
apiKey	Yes	Enter the SAP ARIBA server API key. **Default value**: null

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

Parameter	Mandatory?	Description
username	Yes	Enter the username of the target system that you create for performing connector operations. **Default value**: null
password	Yes	Enter the password of the target system user account that you create for connector operations. **Default value**: null
Connector Server Name	No	This field is blank. If you are using this connector with the Java Conn -ector Server, then provide the name of Connector Server IT Resource here.
XRealm	yes	Enter the Realm Name for which the information has to be fetched. **Default value**: null
partition	yes	Enter the unique partition name for the SAP Ariba solutions which is tenant specific. **Default value**: null
variant	yes	Enter the unique variant name for the SAP Ariba solutions which is tenant specific. **Default value**: null
proxyHost	No	Enter the name of the proxy host used to connect to an external target. **Default value**: null
proxy Password	No	Enter the password for the proxy user
proxy Port	No	Enter the proxy port number. **Default value**: null
proxyUsername	No	Enter the proxy username if you are using a proxy server to access the internet.
SSL	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 table below are mandatory.

Table 3-2 Advanced Settings Parameters

Parameter	Mandatory?	Description
Connector Name	Yes	This entry holds the name of the connector class. **Default value**: org.identityconnectors.sapariba.SAPAribaConnector
Bundle Name	Yes	This entry holds the name of the connector bundle. **Default value**: org.identityconnectors.sapariba
Bundle Version	Yes	This entry holds the version of the connector bundle. **Default value**: 12.3.0
relURIs	Yes	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. **Default value**: "__ACCOUNT__.CREATEOP=$(CreateUserURL)$","__ACCOUNT__.UPDATEOP=$(UpdateUserURL)$","__ACCOUNT__.SEARCHOP=$(UserReconURL)$","__ACCOUNT__.__GROUP__.GROUPLKP=$(GroupLookupURL)$","__ACCOUNT__.ORGANIZATION.Lookup=$(OrganizationLookupURL)$","__ACCOUNT__.CURRENCY.Lookup=$(CurrencyLookupURL)$","__ACCOUNT__.LOCALEID.Lookup=$(LocaleIdLookupURL)$","__ACCOUNT__.__GROUP__.ADDATTRIBUTE=$(GroupsAssignURL)$","__ACCOUNT__.__GROUP__.REMOVEATTRIBUTE=$(GroupsRemoveURL)$"
top	Yes	The number of records to be returned in the response per page. **Default value:**1000

3.3 Attribute Mappings

The following topic provides the attribute mappings details.

- Attribute Mappings for the Target Application

3.3.1 Attribute Mappings for the Target Application

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

The following table lists the user-specific attribute mappings between the process form fields in Oracle Identity Governance and SAP Ariba 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.

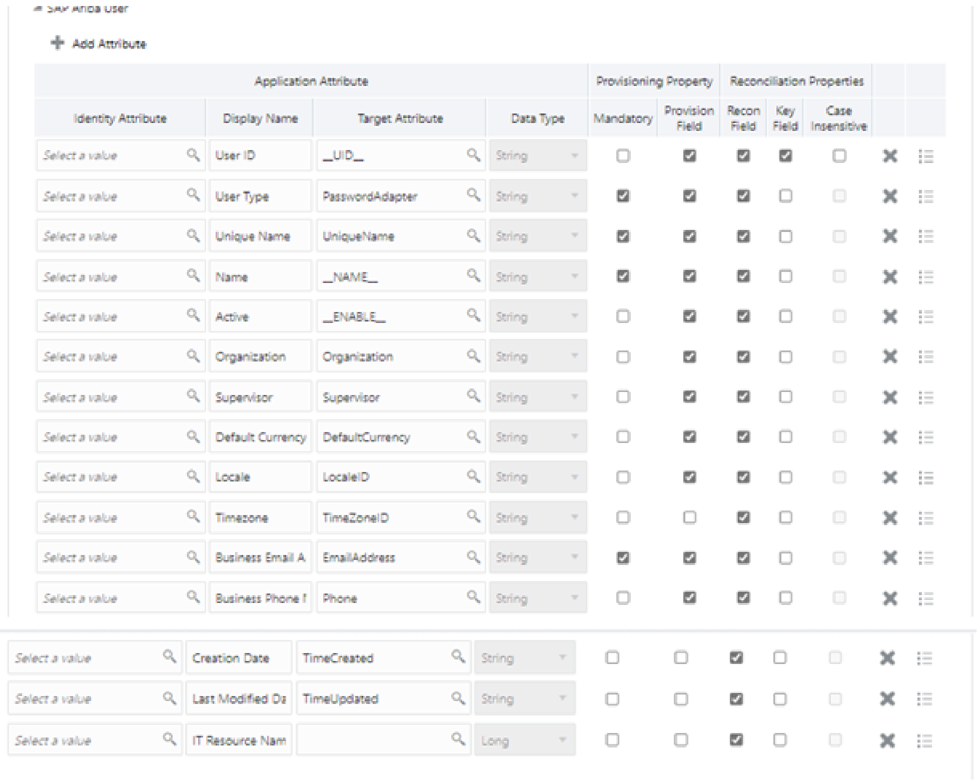
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application Creating a Target Application of *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance*.

Table 3-3 Default Attribute SAP Ariba Target Application

Display Name	Target Attribute	Data Type	Mandatory Provisioning Property?	Provision Field?	Recon Field?	Key Field?	Case Insensitive?
User ID	__UID__	String	No	Yes	Yes	Yes	No
User Type	PasswordAdapter	String	Yes	Yes	Yes	No	Not applicable
Unique Name	UniqueName	String	Yes	Yes	Yes	No	Not applicable
Name	__NAME__	String	Yes	Yes	Yes	No	Not applicable
Active	__ENABLE__	String	No	Yes	Yes	No	Not applicable
Organization	Organization	String	No	Yes	Yes	No	Not applicable
Supervisor	Supervisor	String	No	Yes	Yes	No	Not applicable
Default Currency	DefaultCurrency	String	No	Yes	Yes	No	Not applicable
Locale	LocaleID	String	No	Yes	Yes	No	Not applicable
Timezone	TimeZoneID	String	No	No	Yes	No	Not applicable
Business Email Address	EmailAddress	String	Yes	Yes	Yes	No	Not applicable
Business Phone Number	Phone	String	No	Yes	Yes	No	Not applicable
Creation Date	TimeCreated	String	No	No	Yes	No	Not applicable
Last Modified Date	TimeUpdated	String	No	No	Yes	No	Not applicable
IT Resource Name		Long	No	No	Yes	No	Not applicable

The following figure shows the default User account attribute mappings.

Figure 3-1   Default Attribute Mappings for SAP Ariba User Account



SAP Ariba User

Add Attribute

Application Attribute				Provisioning Property		Reconciliation Properties				
Identity Attribute	Display Name	Target Attribute	Data Type	Mandatory	Provision Field	Recon Field	Key Field	Case Insensitive		
Select a value	User ID	__UID__	String	□	☑	☑	☑	□	✖	☰
Select a value	User Type	PasswordAdapter	String	☑	☑	☑	□	□	✖	☰
Select a value	Unique Name	UniqueName	String	☑	☑	☑	□	□	✖	☰
Select a value	Name	__NAME__	String	☑	☑	☑	□	□	✖	☰
Select a value	Active	__ENABLE__	String	□	☑	☑	□	□	✖	☰
Select a value	Organization	Organization	String	□	☑	☑	□	□	✖	☰
Select a value	Supervisor	Supervisor	String	□	☑	☑	□	□	✖	☰
Select a value	Default Currency	DefaultCurrency	String	□	☑	☑	□	□	✖	☰
Select a value	Locale	LocaleID	String	□	☑	☑	□	□	✖	☰
Select a value	Timezone	TimeZoneID	String	□	□	☑	□	□	✖	☰
Select a value	Business Email A	EmailAddress	String	☑	☑	☑	□	□	✖	☰
Select a value	Business Phone I	Phone	String	□	☑	☑	□	□	✖	☰
Select a value	Creation Date	TimeCreated	String	□	□	☑	□	□	✖	☰
Select a value	Last Modified Da	TimeUpdated	String	□	□	☑	□	□	✖	☰
Select a value	IT Resource Nam		Long	□	□	☑	□	□	✖	☰

SAP Ariba Groups Entitlement

The following table lists the group forms attribute mappings between the process form fields in Oracle Identity Governance and SAP Ariba 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.

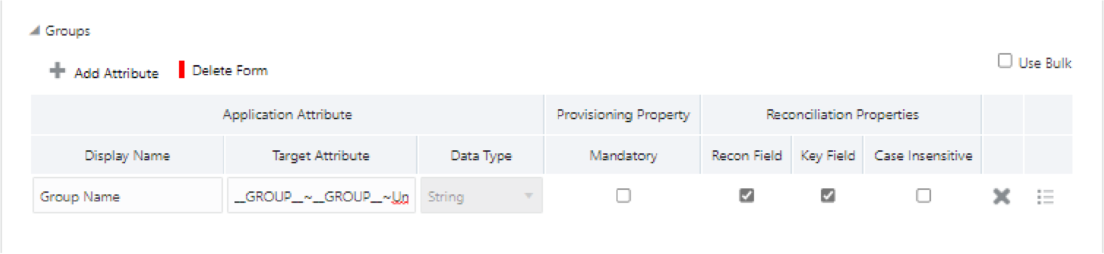
If required, you can edit the default attribute mappings by adding new attributes or deleting existing attributes as described in Creating a Target Application in Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance.

Table 3-4   Default Attribute Mappings for Groups

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

The following figure shows the default Groups Entitlement mapping.

Figure 3-2 Default Attribute Mappings for SAP Ariba Groups



Groups

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	__GROUP__~__GROUP__~Un	String	□	☑	☑	□	✖	☰

3.4 Correlation Rules

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

- Correlation Rules for the Target Application

3.4.1 Correlation Rules for the 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 SAP Ariba 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.

The following table lists the default simple correlation rule for an SAP Ariba connector. If required, you can edit the default correlation rule or add new rules. You can create complex correlation rules also. For more information about adding or editing simple or complex correlation rules, see Updating Identity Correlation Rules in Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance.

Table 3-5 Predefined Identity Correlation Rule for an SAP Ariba Connector

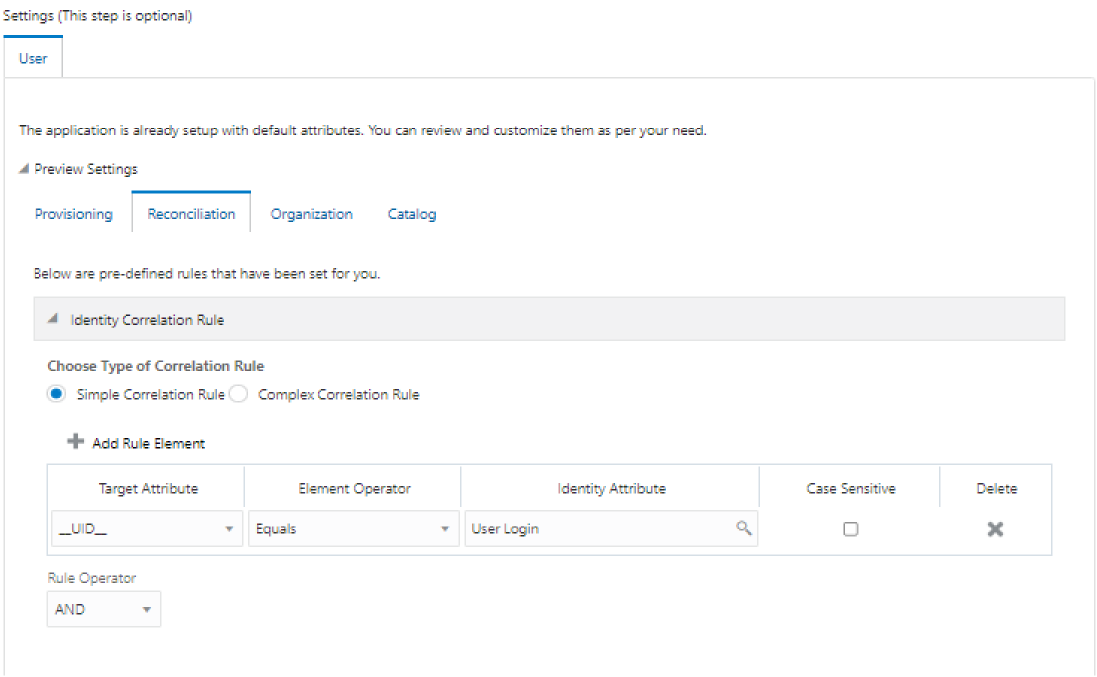
Target Attribute	Element Operator	Identity Attribute	Case Sensitive?
__UID__	Equals	User Login	No

In this identity rule:

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

The following figure shows the Simple Correlation Rule for SAP Ariba Target Application:

Figure 3-3   Simple Correlation Rule for SAP Ariba Target Application



Settings (This step is optional)

User

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

Preview Settings

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

Identity Correlation Rule

Choose Type of Correlation Rule

● Simple Correlation Rule ○ Complex Correlation Rule

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

Rule Operator: AND

Predefined Situations and Responses

The SAP ARIBA 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.

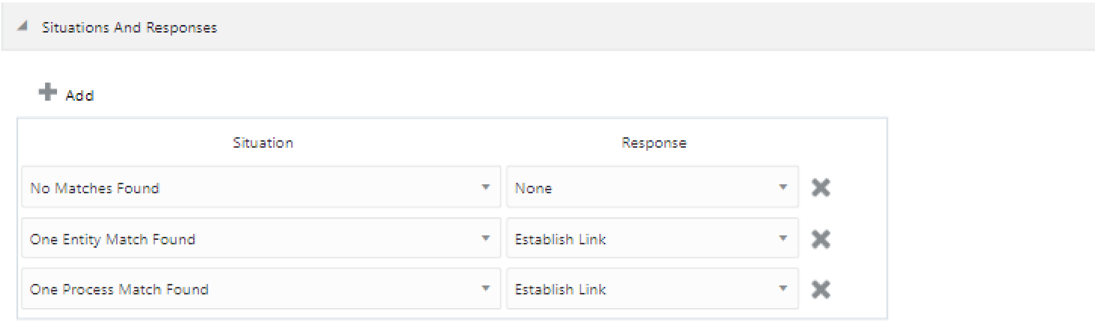
The below table lists the default situations and responses for an SAP Ariba Target application. If required, you can edit these default situations and responses or add new ones. For more information about adding or editing situations and responses, see Updating Situations and Responses Updating Situations and Responses in Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Governance

Table 3-6   Predefined Situations and Responses for an SAP Ariba Target Application

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

The following figure shows the situations and responses for an SAP Ariba that the connector provides by default.

Figure 3-4 Predefined Situations and Responses for an SAP Ariba Application



◢ Situations And Responses

✚ Add

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.

The following reconciliation jobs are available for reconciling user data:

- SAP Ariba Full User Reconciliation: Use this reconciliation job to reconcile user data from a target application.
- SAP Ariba Limited User Reconciliation: Use this reconciliation job to reconcile records from the target system based on a specified filter criterion.
- SAP Ariba Incremental User Reconciliation**:** The SAP Ariba Target Incremental User Reconciliation job is used to fetch the records that are added or modified after the last reconciliation run.

The following table describes the parameters of the SAP Ariba Full User Reconciliation job.

Table 3-7 Parameters of the SAP Ariba 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.
Filter Suffix	Enter the search filter for fetching user records from the target system during a reconciliation run. **Filter Suffix Value:** UniqueName eq ‘<UniqueName>’ For more information about creating filters, see Performing Limited Reconciliation
Object Type	This parameter holds the name of the object type for the reconciliation run. **Default value**: User Do *not* change the default value.

Table 3-7 (Cont.) Parameters of the SAP Ariba Full User Reconciliation Job

Parameter	Description
Scheduled Task Name	Name of the scheduled task used for reconciliation. Do *not* modify the value of this parameter.

Table 3-8 Parameters of the SAP Ariba Incremental 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.
Object Type	This parameter holds the name of the object type for the reconciliation run. **Default value**: User Do *not* change the default value.
Scheduled Task Name	Name of the scheduled task used for reconciliation. Do *not* modify the value of this parameter.
Sync Token	Enter the expression for filtering records that the scheduled job must reconcile. **Sample value**: <String>2023-08-28 14:55:46.118</String>

Reconciliation Jobs for Entitlements

The following jobs are available for reconciling entitlements:

- SAP Ariba Group Lookup Reconciliation
- SAP Ariba Organization Lookup Reconciliation
- SAP Ariba Supervisor Lookup Reconciliation
- SAP Ariba Currency Lookup Reconciliation
- SAP Ariba LocaleID Lookup Reconciliation

The parameters for all the reconciliation jobs are the same.

Table 3-9 Parameters of the Reconciliation Jobs for Entitlements

Parameter	Description
Application Name	Current AOB application name with which the reconciliation job is associated. Do *not* modify this value.

Table 3-9 (Cont.) Parameters of the Reconciliation Jobs for Entitlements

Parameter	Description
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). Depending on the Reconciliation job that you are using, the default values are as follows: • For SAP Ariba Group Lookup Reconciliation: UniqueName • For SAP Ariba Organization Lookup Reconciliation: SystemID • For SAP Ariba Supervisor Lookup Reconciliation: UniqueName • For SAP Ariba Currency Lookup Reconciliation: UniqueName • For SAP Ariba LocaleID Lookup Reconciliation UniqueName Do *not* modify this value.
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_en
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 SAP Ariba Group Lookup Reconciliation: Lookup.SAPAriba.Groups • For SAP Ariba Organization Lookup Reconciliation: Lookup.SAPAriba.organization • For SAP Ariba Supervisor Lookup Reconciliation: Lookup.SAPAriba.supervisor • For SAP Ariba Currency Lookup Reconciliation: Lookup.SAPAriba.currency • For SAP Ariba LocaleID Lookup Reconciliation: Lookup.SAPAriba.localeids 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 SAP Ariba Group Lookup Reconciliation: __GROUP__ • For SAP Ariba Organization Lookup Reconciliation: Organization • For SAP Ariba Supervisor Lookup Reconciliation: Supervisor • For SAP Ariba Currency Lookup Reconciliation: Currency • For SAP Ariba LocaleID Lookup Reconciliation: LocaleID **Note:** Do not change the value of this parameter.

4

Performing Post configuration Tasks for the Connector

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

4.1 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

4.1.1 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*.

4.1.2 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.

4.1.3 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 user credentials and then activate the sandbox that was 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*.

4.1.4 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*

4.2 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 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 Logging on the Connector Server
- Enabling Logging for the Connector Server
- Understanding Log Levels
- Enabling Logging

4.3.1 Understanding Logging on the Connector Server

When you enable logging, the connector server stores in a log file information about events that occur during the course of provisioning and reconciliation operations for different statuses. By default, the connector server logs are set at INFO level and you can change this level to any one of these.

- Error

This level enables logging of information about errors that might allow connector server 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 operation.

- FINE, FINER, FINEST

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

4.3.2 Enabling Logging for the Connector Server

Edit the logging properties file located in the CONNECTOR_SERVER_HOME/Conf directory to enable logging.

1. Open the logging.properties file in a text editor.
2. Navigate to the *CONNECTOR_SERVER_HOME*/Conf directory.
3. Edit the following entry by replacing INFO with the required level of logging:.level=INFO
4. Save and close the file.
5. Restart the connector server.

4.3.3 Understanding Log Levels

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

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

- SEVERE.intValue()+100

This level enables logging of information about fatal errors.

- SEVERE

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

- WARNING

This level enables logging of information about potentially harmful situations.

- INFO

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

- CONFIG

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

- FINE, FINER, FINEST

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

These message types are mapped to ODL message type and level combinations as shown in the following table.

Table 4-1   Log Levels and ODL Message Type: Level Combinations

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

Table 4-1 (Cont.) Log Levels and ODL Message Type: Level Combinations

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

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

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

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

4.3.4 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='SAPARIBA-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> Copy<logger name="
ORG.IDENTITYCONNECTORS.SAPARIBA" level="[LOG_LEVEL]"
useParentHandlers="false"> <handler name="SAPARIBA-handler"/> <handler
name="console-handler"/> </logger>
```

   b. Replace both occurrences of **[LOG_LEVEL]** with the ODL message type and level combination that you require. The Table 4-1 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= ‘SAPARIBA-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\server
s\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.SAPARIBA"
level="NOTIFICATION:1" useParentHandlers="false"> <handler
name="SAPARIBA-handler"/> <handler name="console-handler"/> </logger>
```

2. 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.
3. Save and close the file.
4. Set the following environment variable to redirect the server logs to a file:
    a. For Microsoft Windows: set WLS_REDIRECT_LOG= ***FILENAME***
    b. 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.
5. Restart the application server.

4.4 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, you must create an IT resource for the Connector Server as described in Creating IT Resources of *Oracle Fusion Middleware Administering Oracle Identity Governance.* While creating the IT resource, ensure to select Connector Server from the IT Resource Type list. In addition, 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 4-2 Parameters of the IT Resource for the SAP Ariba Connector

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)

Table 4-2 (Cont.) Parameters of the IT Resource for the SAP Ariba Connector

Parameter	Description
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 the Java Connector Server with SSL for OIG in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Governance*.

4.5 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.xlf
   ```

   > **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 SAP Ariba Application instance. The original code is:

   ```
   <trans-unit id="$
   {adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResourceBund
   le']
   ['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity.userEO
   .UD_ USER_ID__c_description']}"><source>User ID </source><target/></
   trans-unit><trans-unit
   id="sessiondef.oracle.iam.ui.runtime.form.model.RSAForm.entity.SAPAribaF
   ormEO.UD_USER_ID__c_LABEL"><source>User ID</source><target/> </trans-
   unit>
   ```

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

   ```
   global.udf.UD_SAPARIBA_USER_ID = \u30E6\u30FC\u30B6\u30FCID
   ```

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

   ```
   <trans-unit id="$
   {adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResourceBu
   ndle']
   ['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity.use
   rEO.UD_SAPARIBA_USER_ID __c_description']}"><source>User ID</source>
   <target> \u30E6\u30FC\u30B6\u30FCID </target></trans-unit> <trans-
   unitid="sessiondef.oracle.iam.ui.runtime.form.model.Zoom.entity sEO.
   UD_SAPARIBA_USER_ID __c_LABEL"><source>Account Name</source> <target>
   \u30E6\u30FC\u30B6\u30FCID</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.
8. Log out of and log in to Oracle Identity Governance.

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

4.6 Configuring SSL

Configure SSL to secure data communication between Oracle Identity Governance and the SAP Ariba 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 SAP Ariba.
2. Copy the public key certificate of SAP Ariba 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.

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

   -
     ```
     keytool -import -keystore <JAVA_HOME>/jre/lib/security/cacerts -file
     <Cert_Location>/s1ariba.crt -storepass changeit -alias s1ariba1
     ```

   -
     ```
     keytool -import -keystore <WL_HOME>/server/lib/DemoTrust.jks -file
     <Cert_Location>/s1ariba.crt -storepass DemoTrustKeyStorePassPhrase -
     alias s1ariba
     ```

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.
  - In the Oracle Identity Governance cluster, perform this procedure on each node of the cluster and then restart each node.
  - 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

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 (Filtered) 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.

To perform a full reconciliation run, remove (delete) any value currently assigned to the Filter suffix parameters and run one of the reconciliation jobs listed in the Reconciliation Jobs section.

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 timestamp stored in the Sync Token attribute of the scheduled job.

5.1.2 Performing Limited (Filtered) Reconciliation

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

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

This connector provides a Filter Suffix attribute (a scheduled task attribute) that allows you to use any of the attributes of the target system to filter target system records. You specify a value for the Filter Suffix attribute while configuring the user reconciliation scheduled job.

You can perform limited reconciliation by creating filters for the reconciliation module. The connector only supports **UniqueName** filter.

Filter Suffix value: UniqueName eq ‘Alex.Mutu’

Example: UniqueName eq ‘Alex.Mutu’

In this example, the record corresponding to Alex.Mutu is reconciled.

5.2 Configuring Reconciliation Jobs

Configure reconciliation jobs to perform reconciliation runs that check for new information on your target system periodically and replicate 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**.

   > **Note:**
   >
   > If you are using OIG 12cPS4 with 2022OCTBP or later version, log in to Identity Console, click **Manage**, under **System Configuration**, 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:
   a. **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.
   b. **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

You can configure the provisioning operation for the SAP ARIBA connector.

This section provides information on the following topics:

- Guidelines on Performing Provisioning Operations
- Performing Provisioning Operations

5.3.1 Guidelines on Performing Provisioning Operations

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

Provisioning attributes required to create user account.

To create User provisioning operation, follow the following values as required:

- Business Email Address: The user's Business Email Address
- Name: The user's Name
- Unique Name: The user's Unique Name
- User Type: The user's User Type

> **Note:**
>
> Target does not allow to create a user with existing User ID which has already been used in SAP Ariba target.

Attributes required to be updated in the parent form.

- Name: The user's Name
- Default Currency: The user's Default Currency
- Locale: The user's Locale
- Business Email Address: The user's Business Email Address
- Business Phone number: The user's Business Phone number
- Organization: The user's Organization
- Supervisor: The user's Supervisor

5.3.2 Performing Provisioning Operations

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

To perform provisioning operations in Oracle Identity Governance, perform the following steps:

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, 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**.

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 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, Updating the Provisioning Configuration see 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 Files and Directories in the Connector Installation Package

These are the components of the connector installation package that comprise the SAP Ariba connector.

Table 7-1 Files and Directories in the SAP Ariba Connector Installation Package

File in the Installation Package	Description
/bundle/org.identityconnectors.sapariba-12.3.0.jar	This JAR is the ICF connector bundle.
/configuration/SAPARIBA-CI.xml	This XML file contains configuration information.
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/SAPAriba-target-template.xml	This file contains definitions for the connector objects required for creating a Target application. It includes certain details required to connect Oracle Identity Governance with the target system. It also includes configuration details specific to your target system, attribute mappings, correlation rules, and reconciliation jobs.
/xml/SAPAriba-pre-config.xml	This XML file contains definitions for the connector objects associated. it contains definitions of Lookups

Glossary

Index