Oracle® Identity Manager

Connector Guide for ServiceNow

Release 11.1.1

E73592-07

July 2020

Oracle Identity Manager Connector Guide for ServiceNow, Release 11.1.1

E73592-07

Primary Author: Alankrita Prakash

Contributors: Mike Howlett

Contents

Preface

What’s New in Oracle Identity Manager Connector for ServiceNow?

1 About the ServiceNow Connector

2 Deploying the ServiceNow Connector

3 Using the ServiceNow Connector

4 Extending the Functionality of the ServiceNow Connector

5 Known Issues and Workarounds for the ServiceNow Connector

A Files and Directories of the ServiceNow Connector

List of Figures

List of Tables

Preface

This guide describes the connector that is used to integrate Oracle Identity Manager with ServiceNow.

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 Manager, visit the following Oracle Help Center page:

http://docs.oracle.com/cd/E52734_01/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.

Convention	Meaning
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

What’s New in Oracle Identity Manager Connector for ServiceNow?

This chapter provides an overview of the updates made to the software and documentation for the ServiceNow connector in release 11.1.1.5.0.

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

- Software Updates

  This section describes updates made to the connector software. This section also points out the sections of this guide that have been changed in response to each software update.

- Documentation-Specific Updates

  These include major changes made to this guide. For example, the relocation of a section from the second chapter to the third chapter is a documentation-specific update. These changes are not related to software updates.

Software Updates

The following section discusses software updates:

Software Updates in Release 11.1.1.5.0

This is the first release of the Oracle Identity Manager connector for ServiceNow. Therefore, there are no software-specific updates in this release.

Documentation-Specific Updates

The following section discusses documentation-specific updates:

Documentation-Specific Updates in Release 11.1.1.5.0

The following documentation-specific update has been made in revision "07" of this guide:

The "Oracle Identity Governance or Oracle Identity Manager" row of Table 1-1 has been updated.

The following documentation-specific update has been made in revision "06" of this guide:

- The "Oracle Identity Governance or Oracle Identity Manager" row of Table 1-1 has been updated to include support for Oracle Identity Governance release 12c PS4 (12.2.1.4.0).

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

- Sample value for parameter "authenticationServerUrl" has been updated in Table 2-1.

The following documentation-specific update has been made in revision "04" of this guide:

- Sample values for parameters authenticationServerUrl, host, and port of Table 2-1 have been added.

The following documentation-specific update has been made in revision "03" of this guide:

- The “Oracle Identity Manager” row of Table 1-1 has been renamed as "Oracle Identity Governance or Oracle Identity Manager" and also updated for Oracle Identity Governance 12c (12.2.1.3.0) certification.

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

- The “Oracle Identity Manager” row of Table 1-1 has been modified to include support for any later BP in Oracle Identity Manager 11g Release 2 PS3 BP06 (11.1.2.3.6).

1

About the ServiceNow Connector

The ServiceNow connector integrates Oracle Identity Manager (OIM) with the ServiceNow.

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

1.1 Introduction to ServiceNow Connector

Oracle Identity Manager automates access rights management, security, and provisioning of IT resources. ServiceNow connector enables you to use ServiceNow as a managed (target) resource for Oracle Identity Manager.
ServiceNow connector is used to integrate OIM with a ServiceNow instance. ServiceNow connector ensures that all ServiceNow accounts are created, updated, and deactivated on an integrated cycle with the rest of the identity-aware applications in your enterprise. ServiceNow Connector standardizes service processes and implements automation to replace manual tasks.

> **Note:**
>
> At some places in this guide, ServiceNow has been referred to as the **target system**.

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

1.2 Certified Components for ServiceNow Connector

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

Table 1-1 Certified Components

Item	Requirement
Oracle Identity Governance or Oracle Identity Manager	You can use one of the following releases of Oracle Identity Governance of Oracle Identity Manager: • Oracle Identity Governance 12c (12.2.1.4.0) • Oracle Identity Governance 12c (12.2.1.3.0) • Oracle Identity Manager 11*g* Release 2 PS3 BP06 (11.1.2.3.6) and any later BP in this release track
Target System	ServiceNow release Eureka or later
Connector Server	11.1.2.1.0
Connector Server JDK	JDK 1.6 or later

1.3 Certified Languages for the ServiceNow Connector

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

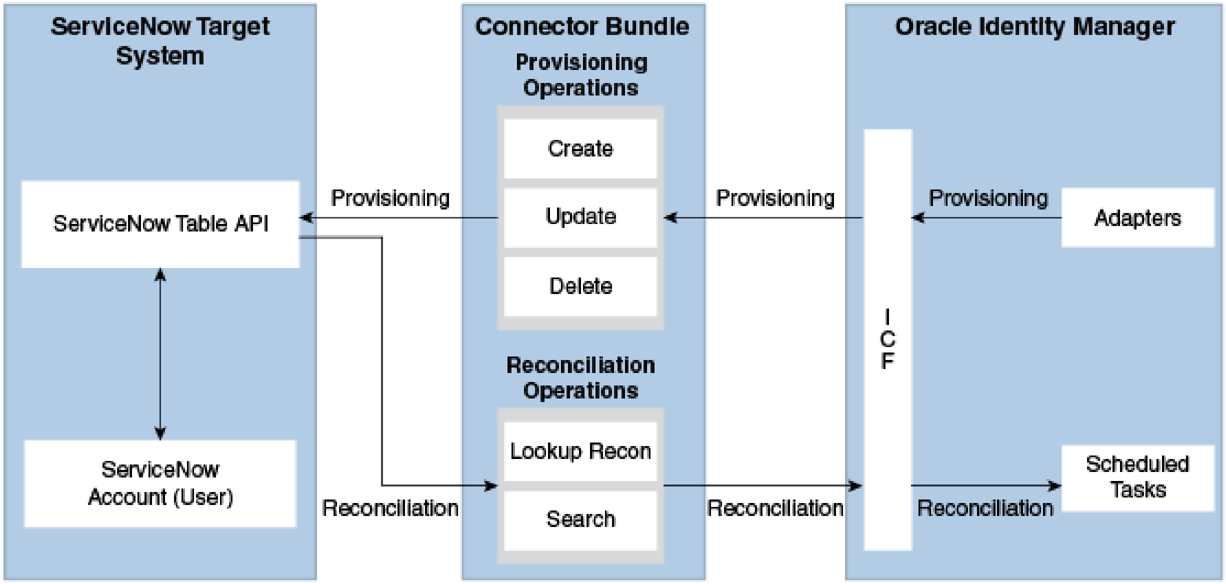
1.4 Architecture of the ServiceNow Connector

The ServiceNow 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 Manager 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 Manager. Therefore, you do not need to configure or modify ICF.

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

Figure 1-1 Connector Architecture



The connector can be 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, updating, or deleting users on the target system through Oracle Identity Manager. During provisioning, the Adapters invoke ICF operation, ICF inturn invokes create operation on the ServiceNow Identity Connector Bundle and then the bundle calls the target system API for provisioning operations. The ServiceNow Table 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 inturn invokes a search operation on the ServiceNow Identity Connector Bundle and then the bundle calls ServiceNow API for reconciliation operation. The API extracts user records that match the reconciliation criteria and hands them over through the bundle and ICF back to the scheduled task, which brings the records to Oracle Identity Manager.

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

The ServiceNow Identity Connector Bundle communicates with the ServiceNow Table API using the HTTPS protocol. The ServiceNow Table API provides programmatic access through REST API endpoints. Apps can use the ServiceNow 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 Manager* for more information about ICF

1.5 Use Cases Supported by the ServiceNow Connector

ServiceNow connector is used to integrate OIM with a ServiceNow instance. ServiceNow connector ensures that all ServiceNow accounts are created, updated, deleted, and deactivated on an integrated cycle with the rest of the identity-aware applications in your enterprise. ServiceNow connector standardizes service processes and implements automation to replace manual tasks. In a typical IT scenario, an organization using OIM wants to manage accounts, user association with a role or with a department across a ServiceNow Cloud instance.

As a business use case, consider a leading logistics company in Australia which was using ServiceNow for the ticketing system solution and OIM for identity management. Before using ServiceNow connector, operations such as create, edit, and delete were performed manually and lacked a centralized streamlining operation. These operations can be easily automated using the ServiceNow REST APIs. By integrating ServiceNow connector with Oracle Identity Manager, the logistics company was able to achieve complete automation.

Following are few example scenarios which ServiceNow connector facilitates:

- **ServiceNow User Management**

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

  ServiceNow connector allows new users to self-provision on a ServiceNow Cloud instance. New users can request and provision from a catalog of cloud-based resources.

  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 create operation against your target system and the user is created on successful execution of the operation. Similarly, operations such as delete and update can be performed.

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

- **Entitlement Grant Management**

  - **ServiceNow Groups**

    In ServiceNow context, a group is a collection of users who share a common purpose. Generally, a group will perform tasks such as approving change requests and resolving incidents.

    For example, consider a network outage scenario. A Network group with several team members will receive a group notification about the incident. The outage incident task can be assigned to any Network group member for a resolution. The ServiceNow connector integration with OIM provides a request-based policy option. Before using ServiceNow connector, the approver must be an user from the Network group. With ServiceNow integration, the said outage resolution can be automatically assigned to users or groups based on predefined polices. For administrators and users, the ServiceNow connector provides an option to facilitate a request-based group membership assignment or group membership revocation options.

  - **ServiceNow Roles**

    In ServiceNow context, a role is an administrator who can create groups and provide access-based permissions to various groups.

    ServiceNow connector manages role memberships. Role memberships provide selective access to ServiceNow functionalities. A user can be a member of one or more roles. Generally, new users are added to a specific role. Each role determines various tasks such as view, update, and delete operations that a ServiceNow user can perform.

As an example, a user with specific role has the rights to view a change request, however does not have access privileges to approve or reject a change request. A ServiceNow user without a role assignment can perform minimal read and write operations. A ServiceNow user needs to have role access privilege in order to create a group. In large organizations, it may be necessary for an administrator to designate other employees to act as administrators to serve different functions. For example, you can set admin roles for your IT staff that can act as support agents to other employees, partners, customers and vendors. With the ServiceNow connector, you can assign or revoke a ServiceNow admin role to users as an entitlement, thus facilitating you to leverage the delegated administration capability of ServiceNow.

1.6 Features of the ServiceNow Connector

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

- Full Reconciliation
- Support for the Connector Server
- Limited (Filtered) Reconciliation
- Transformation and Validation of Account Data

1.6.1 Full Reconciliation

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

After the first full reconciliation run, you can configure your connector for incremental reconciliation.

The connector is capable of performing incremental reconciliation provided the target system supports it. That is, incremental reconciliation is supported if there is a field in the target system that can hold timestamp values.

You can perform a full reconciliation any time.

See Full Reconciliation for more information about performing full reconciliation.

1.6.2 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 Installation for more information about the installation options for this connector.

> **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

1.6.3 Limited (Filtered) Reconciliation

You can reconcile records from the target system based on a specified filter criterion.

You can set a reconciliation filter as the value of the Filter Suffix attribute of the user reconciliation scheduled job. This filter specifies the subset of newly added and modified target system records that must be reconciled. The Filter Suffix attribute helps you to assign filters to the API based on which you will get a filtered response from target.

See Limited (Filtered) Reconciliation for more information about limited reconciliation.

1.6.4 Transformation and Validation of Account Data

You can configure validation of account data that is brought into or sent from Oracle Identity Manager during reconciliation and provisioning.

In addition, you can configure transformation of account data that is brought into Oracle Identity Manager during reconciliation. The following sections provide more information:

- Configuring Transformation of Data During User Reconciliation
- Configuring Validation of Data During Reconciliation and Provisioning

1.7 Lookup Definitions Used During Connector Operations

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

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

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

1.7.1 Lookup Definitions Synchronized with the Target System

During a provisioning operation, you use a lookup field on the process form to select a single value from a set of values. For example, you may want to select a role from the Role Name lookup field to specify the role being assigned to the user. When you deploy the connector, lookup definitions corresponding to the lookup fields on the target system are created in Oracle Identity Manager. **Lookup field synchronization** involves copying additions or changes made to specific fields in the target system to lookup definitions in Oracle Identity Manager.

After you deploy the connector, the following lookup definitions, which are used as an input source for lookup fields, are automatically created in Oracle Identity Manager:

- Lookup.ServiceNow.Groups
- Lookup.ServiceNow.Roles
- Lookup.ServiceNow.Departments

These lookup definitions are empty by default. They are populated with values fetched from the target system when you run the scheduled jobs for lookup field synchronization. For example, when you run the scheduled job for role lookup field synchronization, all Roles on the target system are fetched to Oracle Identity Manager and populated in the Lookup.ServiceNow.Roles lookup definition.

After lookup field synchronization, data in each of the lookup definitions for lookup field synchronization is stored in the following format:

- **Code Key:** *<IT_RESOURCE_KEY>~<FIELD_VALUE_ID>*

  In this format:

  - *IT_RESOURCE_KEY* is the numeric code assigned to an IT resource in Oracle Identity Manager.
  - *FIELD_VALUE_ID* is the ID of the target system field value.

  For example, for the Lookup.ServiceNow.Roles lookup definition, the code key value for one of its entries is `89~1b5d6697-f4a6-4f03-8df7-4fae1512fd16.` In this example, `89` is the numeric code assigned to the IT resource associated with the target system and `1b5d6697-f4a6-4f03-8df7-4fae1512fd16` is the ID of the Role in the target system.

- **Decode:** *<IT_RESOURCE_NAME>~<LOOKUP_FIELD_NAME>*

  In this format:

  - *IT_RESOURCE_NAME* is the name assigned to an IT resource in Oracle Identity Manager.
  - *FIELD_VALUE* is the value of the field in the target system.

  For example, for the Lookup.ServiceNow.Roles lookup definition, the decode value for one of its entries is `ServiceNow~admin.` In this example, `ServiceNow` is the name assigned to the IT resource associated with the target system and `admin` is the name of the Role in the target system.

Table 1-2 shows sample entries in the Lookup.ServiceNow.Groups lookup definition.

Table 1-2 Sample Entries in the Lookup.ServiceNow.Groups Lookup Definition

Code Key	Decode
87~287ebd7da9fe198100f92cc8d1d2154e	ServiceNow~Network
87~287ee6fea9fe198100ada7950d0b1b73	ServiceNow~Database
87~7a0d3844db125200b88df7a0cf96198a	ServiceNow~OIM
87~97f274a1db129200b88df7a0cf9619e2	ServiceNow~OIM Connector

1.7.2 Predefined Lookup Definitions

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

The Preconfigured lookup definitions are as follows:

- Lookup.ServiceNow.Configuration
- Lookup.ServiceNow.UM.Configuration
- Lookup.ServiceNow.UM.ProvAttrMap
- Lookup.ServiceNow.UM.ReconAttrMap
- Lookup.ServiceNow.DateFormat
- Lookup.ServiceNow.Timezone
- Lookup.ServiceNow.CalendarIntegration
- Lookup.ServiceNow.BooleanValues

1.7.2.1 Lookup.ServiceNow.Configuration

The Lookup.ServiceNow.Configuration lookup definition holds connector configuration entries that are used during target resource reconciliation and provisioning operations.

Table 1-3 lists the default entries in the Lookup.ServiceNow.Configuration values lookup definition.

> **Note:**
>
> Do not modify the entries in this lookup definition

Table 1-3 Entries in the Lookup.ServiceNow.Configuration Definition

Code Key	Decode	Description
Any Incremental Recon Attribute Type	true	By default, OIM accepts timestamp information sent from the target system only in Long datatype format. A decode value of True for the Incremental Recon Attribute Type entry indicates that OIM will accept timestamp information in any datatype format.
Bundle Name	org.identityconnectors.servicenow	This entry holds the name of the connector bundle.

Table 1-3 (Cont.) Entries in the Lookup.ServiceNow.Configuration Definition

Code Key	Decode	Description
Bundle Version	1.0.1115	This entry holds the version of the connector bundle.
Connector Name	org.identityconnectors.genericrest.GenericRESTConnector	This entry holds the name of the connector class.

customPayload	"__ACCOUNT__.__GROUP__.UPDATEOP={\"user\": \"$(__UID__)$\",\"group\": \"(sys_id)\"}","
	__ACCOUNT__.__ROLE__.UPDATEOP={\"user\": \"$(__UID__)$\",\"role\": \"(sys_id)\"}","
	__ACCOUNT__.__ENABLE__.UPDATEOP={\"active\":\"$(__ENABLE__)$\",\"locked_out\":\"false\"}"
		This entry lists the payloads for all operations that are not in the standard format.

enableEmptyString	true	This entry holds the boolean value and indicates that an empty string needs to be sent to the target system. When the ServiceNow Table API receives a null value for any parameter, and if the `enableEmptyString` attribute is set to `true`, then an empty string is sent to the target system.
httpHeaderAccept	application/json	This entry holds the accept type expected from the target system in the header.
httpHeaderContentType	application/json	This entry holds the content type expected by the target system in the header.
jsonResourcesTag	"__ACCOUNT__=result","__GROUP__=result", "__ACCOUNT__.__GROUP__=result", "__ACCOUNT__.__ROLE__=result","__ROLE__=result","Department=result", "__ACCOUNT__.__MEMBERSHIP__.__GROUP__=result", "__ACCOUNT__.__MEMBERSHIP__.__ROLE__=result"	This entry holds the json tag value that is used during reconciliation for parsing multiple entries in a single payload.

Table 1-3 (Cont.) Entries in the Lookup.ServiceNow.Configuration Definition

Code Key	Decode	Description
nameAttributes	"__ACCOUNT__.user_name", "__GROUP__.name", "__ROLE__.name","Department.name"	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 user_name.
opTypes	"__ACCOUNT__.__GROUP__.UPDATEOP=POST", "__ACCOUNT__.__ROLE__.UPDATEOP=POST"	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).
passwordAttribute	user_password	This entry holds the name of the target system attribute that is mapped to the __PASSWORD__ attribute of the connector in OIM.

Table 1-3 (Cont.) Entries in the Lookup.ServiceNow.Configuration Definition

Code Key	Decode	Description
relURI’s	"__ACCOUNT__.CREATEOP=/api/now/v1/table/sys_user?sysparm_input_display_value=true", "__ACCOUNT__.SEARCHOP=/api/now/v1/table/sys_user/$(FilterSuffix)$", "__ACCOUNT__=/api/now/v1/table/sys_user/$(__UID__)$?sysparm_input_display_value=true", "__GROUP__.CREATEOP=/api/now/v1/table/sys_user_group", "__GROUP__.SEARCHOP=/api/now/v1/table/sys_user_group/$(FilterSuffix)$", "__GROUP__=/api/now/v1/table/sys_user_group/$(__UID__)$", "__ROLE__.SEARCHOP=/api/now/v1/table/sys_user_role/$(FilterSuffix)$", "__ACCOUNT__.__GROUP__.UPDATEOP=/api/now/table/sys_user_grmember", "__ACCOUNT__.__ROLE__.UPDATEOP=/api/now/table/sys_user_has_role", "__ACCOUNT__.__GROUP__.SEARCHOP=/api/now/v1/table/sys_user_grmember?sysparm_query=user.sys_id=$(__UID__)$", "__ACCOUNT__.__ROLE__.SEARCHOP=/api/now/table/sys_user_has_role?sysparm_query=user.sys_id=$(__UID__)$", "__ACCOUNT__.__GROUP__.DELETEOP=/api/now/table/sys_user_grmember/$(__MEMBERSHIP__.sys_id)$", "__ACCOUNT__.__MEMBERSHIP__.__GROUP__.SEARCHOP =/api/now/v1/table/sys_user_grmember?sysparm_query=user.sys_id=$ (__UID__)$%5E (__UID__)$%5Egroup.sys_id=$(__GROUP__.sys_id) $&sysparm__fields=sys_id", "__ACCOUNT__.__ROLE__.DELETEOP=/api/now/table/sys_user_has_role$(__MEMBERSHIP__.sys_id)$", "__ACCOUNT__.__MEMBERSHIP__.__ROLE__.SEARCHOP=/api/now/v1/table/sys_user_has_role?sysparm_query=user.sys_id=$(__UID__)$%5Erole.sys_id=$(__ROLE__.sys_id) $&sysparm _fields=sys_id", "Department.SEARCHOP =/api/now/v1/table/cmn_department", "__ACCOUNT__.__ENABLE__.UPDATEOP=/api/now/v1/table/sys_user/$(__UID__)$?sysparm__input_display_value=true", "__ACCOUNT__.__ENABLE__.UPDATEOP=/api/now/v1/table/sys_user/$(__UID__)$?sysparm__input_display_value=true"	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. For example, the `"__ACCOUNT__.CREATEOP=/api/now/v1/table/sys_user?sysparm_input_display_value=true"` value implies that `=/api/now/v1/table/sys_user?sysparm_input_display_value=true"` is the relative URL for all create provisioning operations performed on the __ACCOUNT__ object class.

Table 1-3 (Cont.) Entries in the Lookup.ServiceNow.Configuration Definition

Code Key	Decode	Description
specialAttributeHandling	"__ACCOUNT__.__GROUP__.UPDATEOP=SINGLE", "__ACCOUNT__.__ROLE__.UPDATEOP=SINGLE", "__ACCOUNT__.__ENABLE__.CREATEOP=SINGLE", "__ACCOUNT__.__ENABLE__.UPDATEOP=SINGLE"	This entry lists the special attributes whose values should be sent to target 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__.__ENABLE__.CREATEOP` value in decode implies that during an update provisioning operation, the `GROUP` attribute of the `__ACCOUNT__` object class must be sent to the target system one-by-one.
specialAttributeTargetFormat	"__ACCOUNT__.__GROUP__=group", "__ACCOUNT__.__ROLE__=role"	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*
statusAttribute	"__ACCOUNT__.active"	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 `active`.

Table 1-3 (Cont.) Entries in the Lookup.ServiceNow.Configuration Definition

Code Key	Decode	Description
uidAttributes	"__ACCOUNT__.sys_id", "__GROUP__.sys_id", "__ROLE__.sys_id", "Department.sys_id"	This entry holds the uid attribute for all the objects that are handled by this connector. For example, for User accounts, the uid attribute is sys_Id. In other words, the value `__ACCOUNT__.sys_id` in decode implies that the `sys_id` attribute (that is, GUID) of the connector for __ACCOUNT__ object class is mapped to `sys_id` which is the corresponding uid attribute for user accounts in the target system.
User Configuration Lookup	Lookup.ServiceNow.UM.Configuration	This entry holds the name of the lookup definition that contains user-specific configuration properties.

1.7.2.2 Lookup.ServiceNow.UM.Configuration

The Lookup.ServiceNow.UM.Configuration lookup definition contains entries specific to the user object type. This lookup definition is preconfigured and is used during user management operations.

Table 1-4 Entries in the Lookup.ServiceNow.UM.Configuration Lookup Definition

Code Key	Decode
Provisioning Attribute Map	Lookup.ServiceNow.UM.ProvAttrMap
Recon Attribute Map	Lookup.ServiceNow.UM.ReconAttrMap

1.7.2.3 Lookup.ServiceNow.UM.ProvAttrMap

The Lookup.ServiceNow.UM.ProvAttrMap lookup definition holds mappings between process form fields and target system attribute names.

This lookup definition is preconfigured and used during target resource provisioning. Table 1-12 lists the default entries.

You can add entries in this lookup definitions if you want to map new target system attributes for target resource provisioning. See Adding New User Attributes for Provisioning.

1.7.2.4 Lookup.ServiceNow.UM.ReconAttrMap

This lookup definition is preconfigured and used during target resource reconciliation.

The Lookup.ServiceNow.UM.ReconAttrMap lookup definition holds mappings between resource object fields and target system attributes. Table 1-9 lists the default entries.

You can add entries in this lookup definitions if you want to map new target system attributes for target resource reconciliation. See Adding New User Attributes for Reconciliation.

1.7.2.5 Lookup.ServiceNow.DateFormat

The Lookup.ServiceNow.DateFormat lookup definition maps date format values that are used for some of the fields in the target system with the corresponding date format to be displayed in the fields of the OIM User form.

Table 1-5 Entries in the Lookup.ServiceNow.DateFormat Lookup Definition

Code Key (Resource Object Field)	Decode (ServiceNow Field)
dd/MM/yyyy	dd/MM/yyyy
dd.MM.yyyy	dd.MM.yyyy
dd-MM-yyyy	dd-MM-yyyy
MM-dd-yyyy	MM-dd-yyyy
yyyy-MM-dd	yyyy-MM-dd

1.7.2.6 Lookup.ServiceNow.Timezone

The Lookup.ServiceNow.Timezone lookup definition maps timezone values that are used for some of the fields in the target system with the corresponding timezone values to be displayed in the fields of the OIM User form.

Table 1-6 Entries in the Lookup.ServiceNow.Timezone Lookup Definition

Code Key	Decode
GMT	GMT
US/Arizona	US/Arizona
US/Central	US/Central
US/Eastern	US/Eastern

If you want to customize the lookup to add other attributes, you can do so by adding them manually.

1.7.2.7 Lookup.ServiceNow.CalendarIntegration

The Lookup.ServiceNow.CalendarIntegration lookup definition holds information about calender integration that you can select for a target system account that you create through Oracle Identity Manager. This is a static lookup definition.

You must populate the entries of this lookup definition manually. The following is the format of the Code Key and Decode values in this lookup definition:

- **Code Key**: Serial number for calendar integration
- **Decode Key**: Calendar name for integration

Table 1-7 Entries in the Lookup.ServiceNow.CalendarIntegration Lookup Definition

Code Key (Resource Object Field)	Decode (ServiceNow Field)
1	Outlook

1.7.2.8 Lookup.ServiceNow.BooleanValues

The Lookup.ServiceNow.BooleanValues lookup definition maps boolean values that are used for some of the fields in the target system with the corresponding boolean values to be displayed in the fields of the OIM User form.

Table 1-8 lists the default entries in the Lookup.ServiceNow.BooleanValues lookup definition.

Table 1-8 Entries in the Lookup.ServiceNow.BooleanValues Lookup Definition

Code Key (Resource Object Field)	Decode (ServiceNow Field)
true	True
false	False

1.8 Connector Objects Used During Target Resource Reconciliation

Connector objects such as reconciliation rules, reconciliation action rules, and scheduled jobs are used for reconciling user records from the target system into Oracle Identity Manager.

The ServiceNow Target Resource User Reconciliation scheduled job is used to initiate a reconciliation run. This scheduled job is discussed in Reconciliation Scheduled Jobs.

> **See Also:**
>
> Managing Reconciliation in *Oracle Fusion Middleware Administering Oracle Identity Manager* for conceptual information about reconciliation

This section contains the following topics related to connector objects:

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

1.8.1 User Fields for Target Resource Reconciliation

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

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

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

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

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

Code Key (Resource Object Field)	Decode (ServiceNow Field)
Calendar Integration	calendar_integration
Date Format	date_format
Department[LOOKUP]	department.value
Email	email
First Name	first_name
Groups~Group Name[LOOKUP]	__GROUP__~__GROUP__~value
Internal Integration User	internal_integration_user
Last Name	last_name
Locked	locked_out
Mobile Phone	mobile_phone
Password Needs Reset	password_needs_reset
Phone	phone
Roles~Role Name[LOOKUP]	__ROLE__~__ROLE__~value
status	Status __ENABLE__
System Id	__UID__
Time Zone	time_zone
Title	title
User Name	__NAME__
Web Service Access Only	web_service_access_only

1.8.2 Reconciliation Rule for User Target Resource Reconciliation

Reconciliation rules for user target resource reconciliation are used by the reconciliation engine to determine the identity to which Oracle Identity Manager must assign a newly discovered account on the target system.

This section discuss the following topics related to user reconciliation rule for target resource reconciliation:

- Target Resource Reconciliation Rule for Users
- Viewing Reconciliation Rules for Target Resource Reconciliation

1.8.2.1 Target Resource Reconciliation Rule for Users

Reconciliation rules for target resource reconciliation are used by the reconciliation engine to determine the identity to which Oracle Identity Manager must assign a newly discovered account on the target system.

The following is the process-matching rule for users:

Rule name: ServiceNow User Recon Rule

Rule element: User Login Equals User Name

In this rule:

- `User Login` is the User ID field of the OIM User form.
- `User Name` is the unique login name for user in target system.

1.8.2.2 Viewing Reconciliation Rules for Target Resource Reconciliation

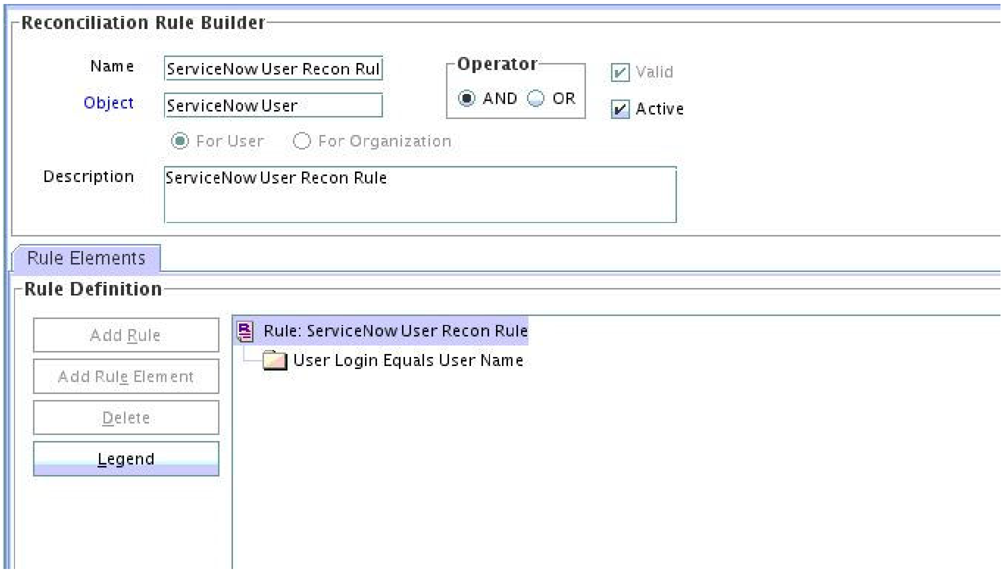
After you deploy the connector, you can view the reconciliation rules on the Reconciliation Rule Builder form in Oracle Identity Manager Design Console.

To view reconciliation rules for target resource reconciliation:

1. Log in to the Oracle Identity Manager Design Console.
2. Expand **Development Tools.**
3. Double-click **Reconciliation Rules.**
4. Search for and open the `ServiceNow User Recon Rule` reconciliation rule.

   Figure 1-2 shows the target resource reconciliation rule for users.

Figure 1-2 Reconciliation Rule for Target Resource Reconciliation of Users



Reconciliation Rule Builder

Name: ServiceNow User Recon Rul

Object: ServiceNow User

● For User ○ For Organization

Description: ServiceNow User Recon Rule

Operator: ● AND ○ OR

☑ Valid

☑ Active

Rule Elements

Rule Definition

Rule: ServiceNow User Recon Rule

User Login Equals User Name

1.8.3 Reconciliation Action Rules for Target Resource Reconciliation

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

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

- Target Resource Reconciliation Action Rules for Users
- Viewing Reconciliation Action Rules for Target Resource Reconciliation

1.8.3.1 Target Resource Reconciliation Action Rules for Users

Reconciliation action rules specify the actions the connector must perform based on the result of the processing of a reconciliation event. The reconciliation action rules for both users and groups are the same.

Table 1-10 Action Rules for Target Resource Reconciliation of Users

Rule Condition	Action
No Matches Found	None
One Entity Match Found	Establish Link
One Process Match Found	Establish Link

1.8.3.2 Viewing Reconciliation Action Rules for Target Resource Reconciliation

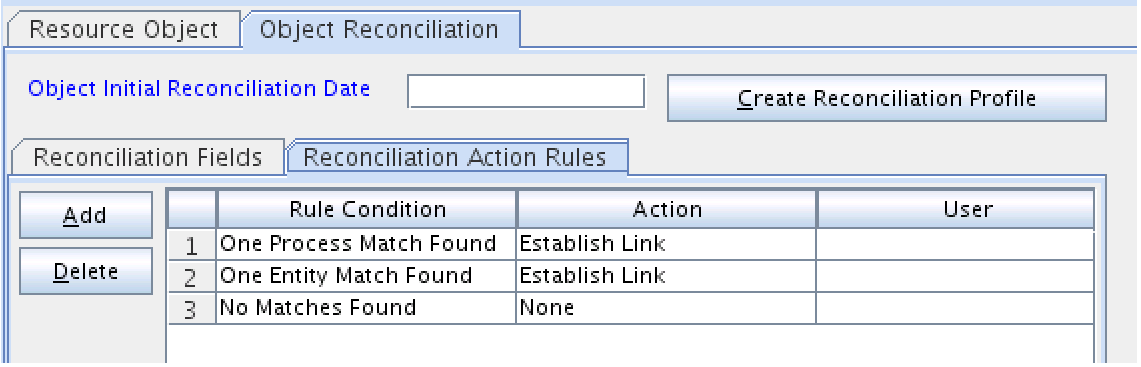
You can view reconciliation action rules on the Object Reconciliation tab of a resource object in Oracle Identity Manager Design Console.

To view reconciliation action rules for target resource reconciliation:

1. Log in to the Oracle Identity Manager Design Console.
2. Expand **Resource Management.**
3. Double-click **Resource Objects.**
4. Search for and open the **ServiceNow User** resource object.
5. Click the **Object Reconciliation** tab, and then click the **Reconciliation Action Rules** tab.

   The Reconciliation Action Rules tab displays the action rules defined for this connector. Figure 1-3 shows the reconciliation action rule for target resource reconciliation.

Figure 1-3 Reconciliation Action Rule for Target Resource Reconciliation


Resource Object
Object Reconciliation
Object Initial Reconciliation Date
Create Reconciliation Profile
Reconciliation Fields
Reconciliation Action Rules
Add
Delete

	Rule Condition	Action	User
1	One Process Match Found	Establish Link	
2	One Entity Match Found	Establish Link	
3	No Matches Found	None	

1.9 Connector Objects Used During Provisioning

Connector objects such as adapters are used for performing provisioning operations on the target system. These adapters perform provisioning functions on the fields defined in the lookup definition for provisioning.

> **See Also:**
>
> Managing Provisioning Tasks in *Oracle Fusion Middleware Performing Self Service Tasks with Oracle Identity Manager* for conceptual information about provisioning

This section contains the following topics:

- Provisioning Functions
- User Fields for Provisioning

1.9.1 Provisioning Functions

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

The Adapter column in Table 1-11 provides the name of the adapter that is used when the function is performed.

See Also:

Types of Adapters in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager* for generic information about process tasks and adapters

Table 1-11 User Provisioning Functions

Function	Adapter
ServiceNow Add Child Object	adpSERVICENOWADDCHILDOBJECT
ServiceNow Bulk Adapter	adpSERVICENOWBULKADAPTER
ServiceNow Create Object	adpSERVICENOWCREATEOBJECT
ServiceNow Delete Object	adpSERVICENOWDELETEOBJECT
ServiceNow Disable Object	adpSERVICENOWDISABLEOBJECT
ServiceNow Enable Object	adpSERVICENOWENABLEOBJECT
ServiceNow Remove Child Object	adpSERVICENOWREMOVECHILDOBJECT
ServiceNow Update Child Data	adpSERVICENOWUPDATECHILDDATA
ServiceNow Update Object	adpSERVICENOWUPDATEOBJECT

1.9.2 User Fields for Provisioning

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

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

Code Key: Name of the process form field

Decode: Name of the target system attribute.

Table 1-12 lists the entries in this lookup definition.

Table 1-12 Entries in the Lookup.ServiceNow.UM.ProvAttrMap Lookup Definition

Code Key	Decode
Calendar Integration	calendar_integration
Date Format	date_format
Department[LOOKUP]	department
Email	email
First Name	first_name
Internal Integration User	internal_integration_user
Last Name	last_name
Locked	locked_out
Mobile Phone	mobile_phone

Table 1-12 (Cont.) Entries in the Lookup.ServiceNow.UM.ProvAttrMap Lookup Definition

Code Key	Decode
Password	__PASSWORD__
Password Needs Reset	password_needs_reset
Phone	phone
System Id	__UID__
Time Zone	time_zone
Title	title
UD_SN_UGP~Group Name[LOOKUP]	__GROUP__~__GROUP__~sys_id
UD_SN_URO~Role Name[LOOKUP]	__ROLE__~__ROLE__~sys_id
User Name	__NAME__
Web Service Access Only	web_service_access_only

1.10 Roadmap for Deploying and Using the Connector

This is the organization of information available in this guide for deploying and using the connector.

The rest of this guide is divided into the following chapters:

- Deploying the ServiceNow Connector describes procedures that you must perform on Oracle Identity Manager and the target system during each stage of connector deployment.
- Using the ServiceNow Connector describes guidelines on using the connector and the procedure to configure reconciliation runs and perform provisioning operations.
- Extending the Functionality of the ServiceNow Connector describes the procedures to perform if you want to extend the functionality of the connector.
- Files and Directories of the ServiceNow Connector lists the files and directories that comprise the connector installation media.

2

Deploying the ServiceNow Connector

The procedure to deploy the connector can be divided into the following stages:

- Registering the Client Application
- Installation
- Postinstallation

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

2.1 Registering the Client Application

Registering a client application with the target system so that the connector can access ServiceNow REST APIs. It also involves creating a user account, modifying ACL values and adding a specific role to a user.

Registering the client application involves performing the following tasks on the target system:

> **Note:**
>
> The detailed instructions for performing these preinstallation tasks are available in the ServiceNow product documentation at https://docs.servicenow.com.

1. Create a user account on the target system and assign the **user_admin** role. The connector uses this account to connect to the target system during reconciliation and provisioning operations.
2. Modify the access control list values (also referred as ACL values) for user role management. This step elevates the user access privilege for the target system user account earlier created. Edit the ACL values for adding various user specific roles that are required for the target system user account.
3. Register the ServiceNow connector as a client application with the ServiceNow instance to provide secure sign-in and authorization for your services. To do so:
   a. Activate the OAuth 2.0 plugin in the ServiceNow instance. This step is required for generating the client ID and client secret values.
   b. Create an OAuth application to generate the client ID and client secret. Note down the client ID and client secret values as they are required while configuring IT resource parameters.

2.2 Installation

Installing the connector requires you to run the connector installer and then configure the IT resource.

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

2.2.1 Understanding Installation of the ServiceNow 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.

2.2.2 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 Connector Installer feature of Oracle Identity System Administration.

To run the Connector Installer:

1. Copy the contents of the connector installation media into the following directory: OIM_HOME/server/ConnectorDefaultDirectory.
2. Log in to Oracle Identity System Administration by using the user account described in Creating the User Account for Installing Connectors of *Oracle Fusion Middleware Administering Oracle Identity Manager*.
3. In the left pane, under Provisioning Configuration, click **Manage Connector**.

4. In the Manage Connector page, click **Install.**
5. From the Connector List, select **ServiceNow Connector -*RELEASE_NUMBER* .**

   This list displays the names and release numbers of connectors whose installation files you copy into the default connector installation directory mentioned in the 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, click **Refresh**.

   c. From the Connector List, select **ServiceNow 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 (Using Deployment Manager)

   c. Compilation of Adapter Definitions

   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. Depending on the reason for the failure, make the required correction and then perform one of the following steps:

   - Retry the installation by clicking **Retry.**
   - Cancel the installation and begin again from Step 1.

   If all three tasks of the connector installation process are successful, then a message indicating successful installation is displayed.
8. Click **Exit** to close the installation page.

   If all three tasks of the connector installation process are successful, then a message indicating successful installation is displayed. In addition, a list of steps that you must perform after the installation are displayed. These steps are as follows:

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

   b. Configuring the IT resource for the connector.

      The procedure to configure the IT resource is described later in this guide

   c. Configuring the scheduled jobs.

      The procedure to configure the IT resource is described later in this guide. When you run the Connector Installer, it copies the connector files to destination directories on the Oracle Identity Manager host computer. Connector files available as part of the connector installation media are listed in Files and Directories of the ServiceNow Connector.

2.2.3 Configuring the IT Resource for the Target System

An IT resource for your target system is created after you install the connector. You configure this IT resource to let the connector connect Oracle Identity Manager with your target system.

This section discusses the following topics:

- IT Resource Parameters
- Specifying Values for the IT Resource Parameters

2.2.3.1 IT Resource Parameters

An IT resource is composed of parameters that store connection and other generic information about a target system. Oracle Identity Manager uses this information to connect to a specific installation or instance of your target system.

Table 2-1 IT Resource Parameters

Parameter	Description
Configuration Lookup	Name of the lookup definition that stores configuration information used during reconciliation and provisioning. Default value: `Lookup.ServiceNow.Configuration`
Connector Server Name	If you have deployed the ServiceNow connector in the Connector Server, then enter the name of the IT resource for the Connector Server.
authenticationServerUrl	Enter the URL of the authentication server that is used to authenticate the resource owner user name and password. Sample value: https://ven01622.service-now.com/`oauth_token.do`
authenticationType	Type of the authentication. Default value: `password` Do not modify the value of the parameter. **Note**: The sample value implies OAuth 2.0 resource owner password. ServiceNow target instance supports only the OAuth 2.0 resource owner password type.
clientId	Client identifier issued to the client during the registration process. Sample value: `ab0781d7c00a120039f0dbb350692319` The clientId is obtained while performing the procedure described in Registering the Client Application.

Table 2-1 (Cont.) IT Resource Parameters

Parameter	Description
clientSecret	Enter the client secret used to authenticate the identity of the client application. Sample value:`?*AV79Zx}` The clientSecret is obtained while performing the procedure described in Registering the Client Application.
host	Host name or IP address of the computer hosting the target system. Sample value: `ven01623.service-now.com`
password	Password used for the OAuth 2.0 resource owner password authentication.
port	Port number at which the target system is listening. Sample value: `443`
sslEnabled	If the target system requires SSL connectivity, then set the value of this parameter to `true`. Otherwise set the value to `false`.
username	Enter the user name for the OAuth 2.0 authentication. This user name is used during password authentication of the resource owner. Sample value:`johnsmith`
proxyHost	Name of the proxy host used to connect to an external target. Sample value: `www.example.com.`
proxyPort	Proxy port number. Sample value: `80`
proxyUser	Proxy user name of the target system user account that Oracle Identity Manager uses to connect to the target system.
proxyPassword	Password of the proxy user ID of the target system user account that Oracle Identity Manager uses to connect to the target.

2.2.3.2 Specifying Values for the IT Resource Parameters

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

The ServiceNow IT resource is automatically created when you run the Connector Installer. You must specify values for the parameters as follows:

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 `ServiceNow` 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, ServiceNow. See IT Resource Parameters for information about IT resource parameters.
7. To save the values, click **Update**.

2.3 Postinstallation

This topic discusses the following postinstallation procedures:

- Configuring Oracle Identity Manager
- Localizing Field Labels in UI Forms
- Clearing Content Related to Connector Resource Bundles from the Server Cache
- Managing Logging for the ServiceNow Connector
- Configuring SSL for ServiceNow

2.3.1 Configuring Oracle Identity Manager

You must create a UI form and an application instance for the resource against which you want to perform reconciliation and provisioning operations. In addition, you must run entitlement and catalog synchronization jobs.

These procedures are described in the following sections:

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

2.3.1.1 Creating and Activating a Sandbox

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 and Deactivating a Sandbox in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager*.

2.3.1.2 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 Manager*.

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

2.3.1.3 Associating the Form with the Application Instance

By default, an application instance named ServiceNow Application Instance is automatically created after you install the connector. You must associate this application instance with the form created in Creating a New UI Form.

After updating the application instance, you must publish it to an organization to make the application instance available for requesting and subsequent provisioning to users. However, as a best practice, perform the following procedure before publishing the application instance:

1. In Oracle Identity System Administration, deactivate the sandbox.
2. Log out of Oracle Identity System Administration.
3. Log in to the Oracle Identity Self Service and activate the sandbox that you deactivated in Step 1.
4. In the Catalog page, search for and add to cart the application instance updated in and then click **Checkout**.
5. Publish the application instance only if everything appears correctly. Otherwise, fix the issues and then publish the application instance. See Publishing an Application Instance to Organizations in *Oracle Fusion Middleware Administering Oracle Identity Manager* .

2.3.1.4 Publishing a Sandbox

Before you publish a sandbox, perform the following procedure as a best practice to validate all sandbox changes made till this stage as it is hard to revert changes once a sandbox is published:

1. In the Oracle Identity System Administration, deactivate the sandbox.
2. Log out of the Oracle Identity System Administration.
3. Log in to the Oracle 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 ServiceNow application instance form appears with correct fields.
5. Publish the sandbox. See Publishing a Sandbox in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager*.

2.3.1.5 Harvesting Entitlements and Sync Catalog

To harvest entitlements and sync catalog:

1. Run the scheduled jobs for lookup field synchronization discussed in Scheduled Job for Lookup Field Synchronization.
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 Manager*

2.3.1.6 Updating an Existing Application Instance with a New Form

For any changes you do in the Form Designer, you must create a new UI form and update the changes in an application instance. To update an existing application instance with a new form:

1. Create and activate a sandbox. See Creating a Sandbox and Activating and Deactivating a Sandbox in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager*.
2. Create a new UI form for the resource. See Creating Forms By Using the Form Designer in *Oracle Fusion Middleware Administering Oracle Identity Manager*.
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 Publishing a Sandbox in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager*.

2.3.2 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 label 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 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_en.xlf
6. Edit the BizEditorBundle.xlf file in the following manner:
   a. Search for the following text:

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

Replace with the following text:

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

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

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

c. Search for the application instance code. This procedure shows a sample edit for ServiceNow 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_SN_USR_USERNAME__c_description']}">
<source>User Name</source>
<target/>
</trans-unit>
<trans-unit
id="sessiondef.oracle.iam.ui.runtime.form.model.snform.entity.snf
ormEO.UD_SN_USR_USERNAME__c_LABEL">
<source>First Name</source>
<target/>
</trans-unit>
```

d. Open the properties file from resource folder in the connector package, for example `ServiceNow_ja.properties`, and get the value of the attribute from the file, for example,

```
global.udf.UD_SNA_USR_ USER_NAME
=\u30A2\u30AB\u30A6\u30F3\u30C8\u540D
```

e. Replace the original code shown in Step 7.c with the following:

```
<trans-unit id="$
{adfBundle['oracle.adf.businesseditor.model.util.BaseRuntimeResou
rceBundle']
['persdef.sessiondef.oracle.iam.ui.runtime.form.model.user.entity
.use
rEO.UD_SN_USR_ USER_NAME __c_description']}">
<source>Account Name</source>
<target>u30A2\u30AB\u30A6\u30F3\u30C8\u540D</target>
</trans-unit>
<trans-unit
```

```
id="sessiondef.oracle.iam.ui.runtime.form.model.Servicenow.entity
sEO.UD_SN_USR_UserName__c_LABEL">
<source>Account Name</source>
<target>\u30A2\u30AB\u30A6\u30F3\u30C8\u540D</target>
</trans-unit>
```

f. Repeat Steps 7.a through 7.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.

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

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

i. Log out of and log in to Oracle Identity Manager.

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

When you deploy the connector, the resource bundles are copied from the resources directory on the installation media into the Oracle Identity Manager database. Whenever you add a new resource bundle to the connectorResources directory or make a change in an existing resource bundle, you must clear content related to connector resource bundles from the server cache.

To clear content related to connector resource bundles from the server cache you can either restart Oracle Identity Manager or run the PurgeCache utility. The following is the procedure to clear the server cache by running the PurgeCache utility:

1. In a command window, switch to the *OIM_HOME*/server/bin directory.
2. Enter one of the following commands:
   - On Microsoft Windows: `PurgeCache.bat All`
   - On UNIX: `PurgeCache.sh All`

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

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

   In this format:

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

You can use the PurgeCache utility to purge the cache for any content category.

2.3.4 Managing Logging for the ServiceNow Connector

You can set a log level based on Oracle Java Diagnostic Logging and enable logging in the Oracle WebLogic Server. The following topics contain detailed information:

- Understanding Log Levels
- Enabling Logging

2.3.4.1 Understanding Log Levels

Oracle Identity Manager uses Oracle Java Diagnostic Logging (OJDL) for logging. OJDL is based on java.util.logger. To specify the type of event for which you want logging to take place, you can set the logs to one of the following available levels:

- 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 log levels are mapped to ODL message type and level combinations as shown in Table 2-2.

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

Log 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

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

Log Level	ODL Message Type:Level
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 Manager.

2.3.4.2 Enabling Logging

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

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

   b. Replace both occurrences of **`[LOG_LEVEL]`** with the ODL message type and level combination that you require. .

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

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

```
<log_handler name='Servicenow-handler' level='NOTIFICATION:1'
class='oracle.core.ojdl.logging.ODLHandlerFactory'>
<property name='logreader:' value='off'/>
     <property name='path' value='/<%OIM_DOMAIN%>/servers/oim_server1/
logs/serviceNowScriptLogs.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.SERVICENOW" level="NOTIFICATION:1"
useParentHandlers="false">
     <handler name="Servicenow-handler"/>
     <handler name="console-handler"/>
   </logger>
```

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

2. Save and close the file.
3. Set the following environment variable to redirect the server logs to a file:

   For Microsoft Windows:

   ```
   set WLS_REDIRECT_LOG=FILENAME
   ```

   For UNIX:

   ```
   export WLS_REDIRECT_LOG=FILENAME
   ```

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

2.3.5 Configuring SSL for ServiceNow

Configure SSL to secure data communication between Oracle Identity Manager and ServiceNow.

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

   ```
   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 Manager is in sync with the validity date of the SSL certificate to avoid any errors during SSL communication.

3 Using the ServiceNow Connector

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

This section discusses the following topics:

- Scheduled Job for Lookup Field Synchronization
- Configuring Reconciliation for ServiceNow Connector
- Configuring Scheduled Jobs
- Guidelines on Performing Provisioning Operations
- Performing Provisioning Operations
- Uninstalling the Connector

3.1 Scheduled Job for Lookup Field Synchronization

Scheduled jobs for lookup field synchronization fetch the most recent values from specific fields in the target system to lookup definitions in Oracle Identity Manager. These lookup definitions are used as an input source for lookup fields in Oracle Identity Manager.

The following scheduled jobs are used for lookup fields synchronization:

- ServiceNow Group Lookup Recon
- ServiceNow Role Lookup Recon
- ServiceNow Department Lookup Recon

Values fetched by these scheduled jobs from the target system are populated in the Lookup.ServiceNow.Groups, Lookup.ServiceNow.Roles and Lookup.ServiceNow.Departments respectively. The attributes for all the scheduled jobs for lookup field synchronization are the same.Table 3-1 describes the attributes of the scheduled jobs. The procedure to configure scheduled jobs is described later in this guide.

Table 3-1 Attributes of the Scheduled Job for Lookup Field Synchronization

Attribute	Description
Code Key Attribute	Enter the name of the 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__

Table 3-1 (Cont.) Attributes of the Scheduled Job for Lookup Field Synchronization

Attribute	Description
Decode Attribute	Enter the name of the 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__
IT Resource Name	Name of the IT resource for the target system installation from which you reconcile user records. Default value: `ServiceNow`
Lookup Name	Name of the lookup definition in Oracle Identity Manager that must be populated with values fetched from the target system. Depending on the scheduled job you are using, the default values are as follows: • For ServiceNow Group Lookup Recon Scheduled Job: `Lookup.ServiceNow.Groups` • For ServiceNow Department Lookup Recon Scheduled Job: `Lookup.ServiceNow.Department` • For ServiceNow Role Lookup Recon Scheduled Job: `Lookup.ServiceNow.Roles`

Object Type	Name of the type of object you want to reconcile.
	Depending on the scheduled job you are using, the default values are as follows:
	• For ServiceNow Group Lookup Recon Scheduled Job: `_GROUP_`
	• For ServiceNow Department Lookup Recon Scheduled Job:
	`Department`
	• For ServiceNow Role Lookup Recon Scheduled Job:
	`_ROLE_`

3.2 Configuring Reconciliation for ServiceNow Connector

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

This section discusses the following topics related to configuring reconciliation:

- Full Reconciliation
- Limited (Filtered) Reconciliation
- Reconciling Large Number of Records
- Reconciliation Scheduled Jobs

3.2.1 Full Reconciliation

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

After you deploy the connector, you must first perform full reconciliation. To perform a full reconciliation run, ensure that no value is specified for the Filter Suffix attribute of the scheduled job for reconciling users and groups. If the target system contains more number of records than what it can return in a single response, then use the Flat File connector to perform full reconciliation. See Reconciling Large Number of Records.

3.2.2 Limited (Filtered) 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 `sys_id`. The `sys_id` attribute is present in the target system and OIM. Filtered reconciliation is performed using the `sys_id` as a filter suffix attribute.

> **Note:**
>
> In the current connector release, the `sys_id` attribute is the only filter suffix supported for filtering records.

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 `sys_id` attribute of the target system to filter target system records. The `sys_id` 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 `/0e220301db039a00b88df7a0cf9619`. The value provided in the filter suffix parameter varies in accordance with the target system. See Reconciliation Scheduled Jobs.

3.2.3 Reconciling Large Number of Records

During a reconciliation run, if the target system contains more number of records than what it can return in a single response, then you must use the Flat File connector to fetch all the records into Oracle Identity Manager.

To reconcile a large number of records from the target system into Oracle Identity Manager:

1. Export all users in the target system to a flat file.
2. Copy the flat file to a location that is accessible from Oracle Identity Manager.
3. Create a schema file representing the structure of the flat file.

4. Install the Flat File connector.
5. Configure the Flat File IT resource.
6. If you want to perform trusted source reconciliation, then configure and run the Flat File Users Loader scheduled job.

   While configuring this scheduled job, ensure that you set the value of the **Target IT Resource Name** attribute to `ServiceNow` and **Target Resource Object Name** to `ServiceNow User`.
7. If you want to perform target resource reconciliation, then configure and run the Flat File Accounts Loader scheduled job.

   While configuring this scheduled job, ensure that you set the value of the **Target IT Resource Name** attribute to `ServiceNow` and **Target Resource Object Name** to `ServiceNow User .`

3.2.4 Reconciliation Scheduled Jobs

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

In ServiceNow connector, the Scheduled Job for Reconciliation of User Records is automatically created in the Oracle Identity Manager. The Scheduled Job for Reconciliation of User Records is used to reconcile user data in the target resource (account management) mode of the connector.

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

Table 3-2 Attributes of the User Reconciliation Scheduled Job

Attribute	Description
Filter Suffix	Enter the search filter for fetching records from the target system during a reconciliation run. Sample value: `/0e220301db039a00b88df7a0cf9619` See Limited (Filtered) Reconciliation.
Latest Token	The Latest Token attribute is used for internal purposes. By default, this value is empty. Do not enter a value for this attribute. The reconciliation engine automatically enters a value in this attribute. Sample value: 1354753427000
IT Resource Name	Name of the IT resource for the target system installation from which you want to reconcile user records. Default value: `ServiceNow`
Object Type	Type of object you want to reconcile. Default value: `User` **Note:** User is the only object that is supported. Therefore, do not change the value of this attribute.

Table 3-2 (Cont.) Attributes of the User Reconciliation Scheduled Job

Attribute	Description
Resource Object Name	Name of the resource object that is used for reconciliation. Default value: `ServiceNow User` **Note**: Do not change the value of this attribute
Scheduled Task Name	Name of the scheduled task that is used for reconciliation. Default value: `ServiceNow User Reconciliation Test`

3.3 Configuring Scheduled Jobs

You must configure and run scheduled jobs to perform a reconciliation run.

To configure a scheduled job:

1. Log in to Oracle Identity System Administration.
2. In the left pane, under System Management, click **Scheduler.**
3. Search for and open the scheduled task as follows:
   a. On the left pane, in the Search field, enter the name of the scheduled job as the search criterion. Alternatively, you can click **Advanced Search** and specify the search criterion.
   b. In the search results table on the left pane, click the scheduled job in the Job Name column.
4. On the Job Details tab, you can modify the following parameters:
   - **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.

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

   In addition to modifying the job details, you can enable or disable a job.
5. On the Job Details tab, in the Parameters region, specify values for the attributes of the scheduled task.

> **Note:**
>
> - Attribute values are predefined in the connector XML file that you import. Specify values only for those attributes that you want to change.
> - Values (either default or user-defined) must be assigned to all the attributes. If even a single attribute value is left empty, then reconciliation is not performed.
> - Attributes of the scheduled job are discussed in Reconciliation Scheduled Jobs.

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

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

3.4 Guidelines on Performing Provisioning Operations

You must apply the below guideline while performing a provisioning operation:

For a Create User provisioning operation, you must specify a value for the User Name field. For example, John Doe. It is a mandatory field, other mandatory fields are Display Name, Password, MailNickname, and Usage Location.

3.5 Performing Provisioning Operations

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

To perform provisioning operations in Oracle Identity Manager:

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

> **Note:**
>
> Ensure to select proper values for lookup type fields as there are a few dependent fields. Selecting a wrong value for such fields may result in provisioning failure.

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

3.6 Uninstalling the Connector

Uninstalling the connector involves deleting data related to the connector from Oracle Identity Manager Database. You use the Uninstall Connectors utility to uninstall a connector.

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

4

Extending the Functionality of the ServiceNow Connector

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

This chapter discusses the following sections:

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

- Adding New User Attributes for Reconciliation
- Adding New User Attributes for Provisioning
- Configuring Validation of Data During Reconciliation and Provisioning
- Configuring Transformation of Data During User Reconciliation
- Configuring the Connector for Multiple Installations of the Target System
- Defining the Connector

4.1 Adding New User Attributes for Reconciliation

The connector provides a default set of attribute mappings for reconciliation between Oracle Identity Manager and the target system. If required, you can add new user attributes for reconciliation.

The default attribute mappings for reconciliation are listed in Table 1-9.

> **Note:**
>
> This connector supports configuration of already existing (standard) attributes of ServiceNow for reconciliation.

The following topics discuss the procedure to add new attributes for users:

- Adding New Attributes on the Process Form
- Adding Attributes to the Resource Object

- Creating Reconciliation Field Mapping
- Creating Entries in Lookup Definition for Reconciliation
- Performing Changes in a New UI Form

4.1.1 Adding New Attributes on the Process Form

You add a new attribute on the process form in the Form Designer section of Oracle Identity Manager Design Console.

1. Log in to the Oracle Identity Manager Design Console.
2. Expand **Development Tools**, and double-click **Form Designer**.
3. Search for and open the **UD_ServiceNow_USR** process form.
4. Click **Create New Version**, and then click **Add**.
5. Enter the details of the field.

   For example, if you are adding the TELEPHONENUMBER field, enter `UD_ServiceNow_USR_TELEPHONENUMBER` in the Name field and then enter other details such as Variable Type, Length, Field Label, and Field Type.
6. Click the Save icon, and then click **Make Version Active**. The following screenshot shows the new field added to the process form.

Figure 4-1 New Field Added to the Process Form

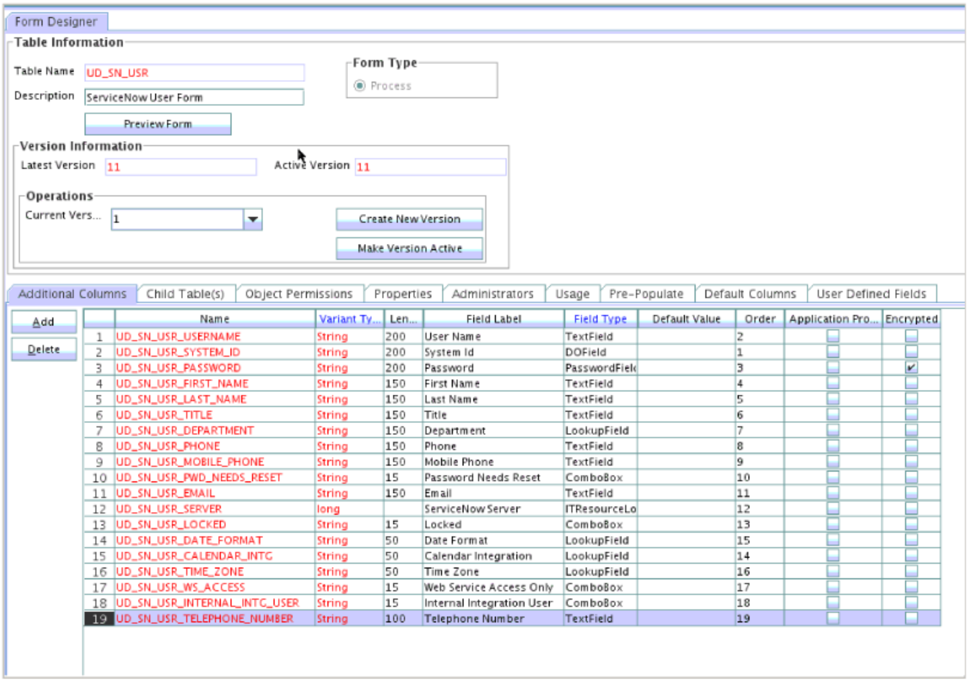


Table Information

Table Name: UD_SN_USR

Description: ServiceNow User Form

Form Type: ● Process

Version Information

Latest Version: 11

Active Version: 11

Operations

Current Vers...: 1

	Name	Variant Ty...	Len...	Field Label	Field Type	Default Value	Order	Application Pro...	Encrypted
1	UD_SN_USR_USERNAME	String	200	User Name	TextField		2	□	□
2	UD_SN_USR_SYSTEM_ID	String	200	System Id	DOField		1	□	□
3	UD_SN_USR_PASSWORD	String	200	Password	PasswordFiel		3	□	☑
4	UD_SN_USR_FIRST_NAME	String	150	First Name	TextField		4	□	□
5	UD_SN_USR_LAST_NAME	String	150	Last Name	TextField		5	□	□
6	UD_SN_USR_TITLE	String	150	Title	TextField		6	□	□
7	UD_SN_USR_DEPARTMENT	String	150	Department	LookupField		7	□	□
8	UD_SN_USR_PHONE	String	150	Phone	TextField		8	□	□
9	UD_SN_USR_MOBILE_PHONE	String	150	Mobile Phone	TextField		9	□	□
10	UD_SN_USR_PWD_NEEDS_RESET	String	15	Password Needs Reset	ComboBox		10	□	□
11	UD_SN_USR_EMAIL	String	150	Email	TextField		11	□	□
12	UD_SN_USR_SERVER	long		ServiceNow Server	ITResourceLo		12	□	□
13	UD_SN_USR_LOCKED	String	15	Locked	ComboBox		13	□	□
14	UD_SN_USR_DATE_FORMAT	String	50	Date Format	LookupField		15	□	□
15	UD_SN_USR_CALENDAR_INTG	String	50	Calendar Integration	LookupField		14	□	□
16	UD_SN_USR_TIME_ZONE	String	50	Time Zone	LookupField		16	□	□
17	UD_SN_USR_WS_ACCESS	String	15	Web Service Access Only	ComboBox		17	□	□
18	UD_SN_USR_INTERNAL_INTC_USER	String	15	Internal Integration User	ComboBox		18	□	□
19	UD_SN_USR_TELEPHONE_NUMBER	String	100	Telephone Number	TextField		19	□	□

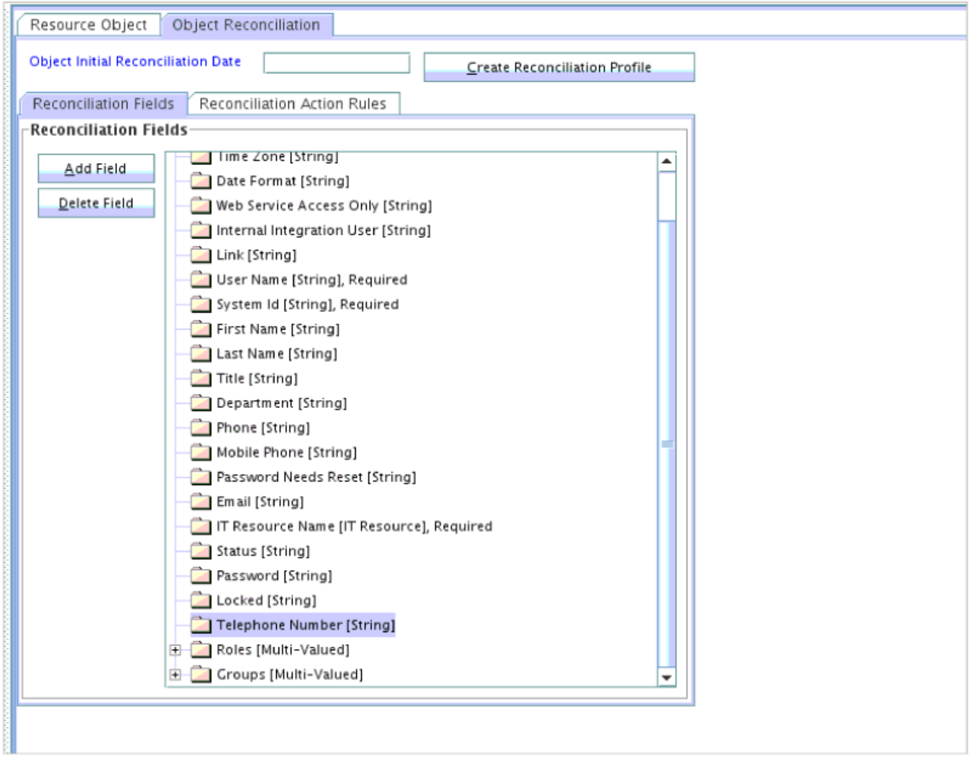
4.1.2 Adding Attributes to the Resource Object

You can add the new attribute to the resource object in the Resource Objects section of Oracle Identity Manager Design Console.

1. Expand **Resource Management**, and double-click **Resource Objects**.
2. Search for and open the **ServiceNow User** resource object.
3. On the Object Reconciliation tab, click **Add Field**.
4. Enter the details of the field.

   For example, enter `TELEPHONE NUMBER` in the **Field Name** field and select **String** from the **Field Type** list. Later in this procedure, you enter the field name as the Code value of the entry that you create in the lookup definition for reconciliation.
5. Click the Save icon. The following screenshot shows the new reconciliation field added to the resource object:

Figure 4-2 New Reconciliation Field Added to the Resource Object



Time Zone [String]
Date Format [String]
Web Service Access Only [String]
Internal Integration User [String]
Link [String]
User Name [String], Required
System Id [String], Required
First Name [String]
Last Name [String]
Title [String]
Department [String]
Phone [String]
Mobile Phone [String]
Password Needs Reset [String]
Email [String]
IT Resource Name [IT Resource], Required
Status [String]
Password [String]
Locked [String]
Telephone Number [String]
Roles [Multi-Valued]
Groups [Multi-Valued]

6. Click **Create Reconciliation Profile**. This copies changes made to the resource object into the MDS.

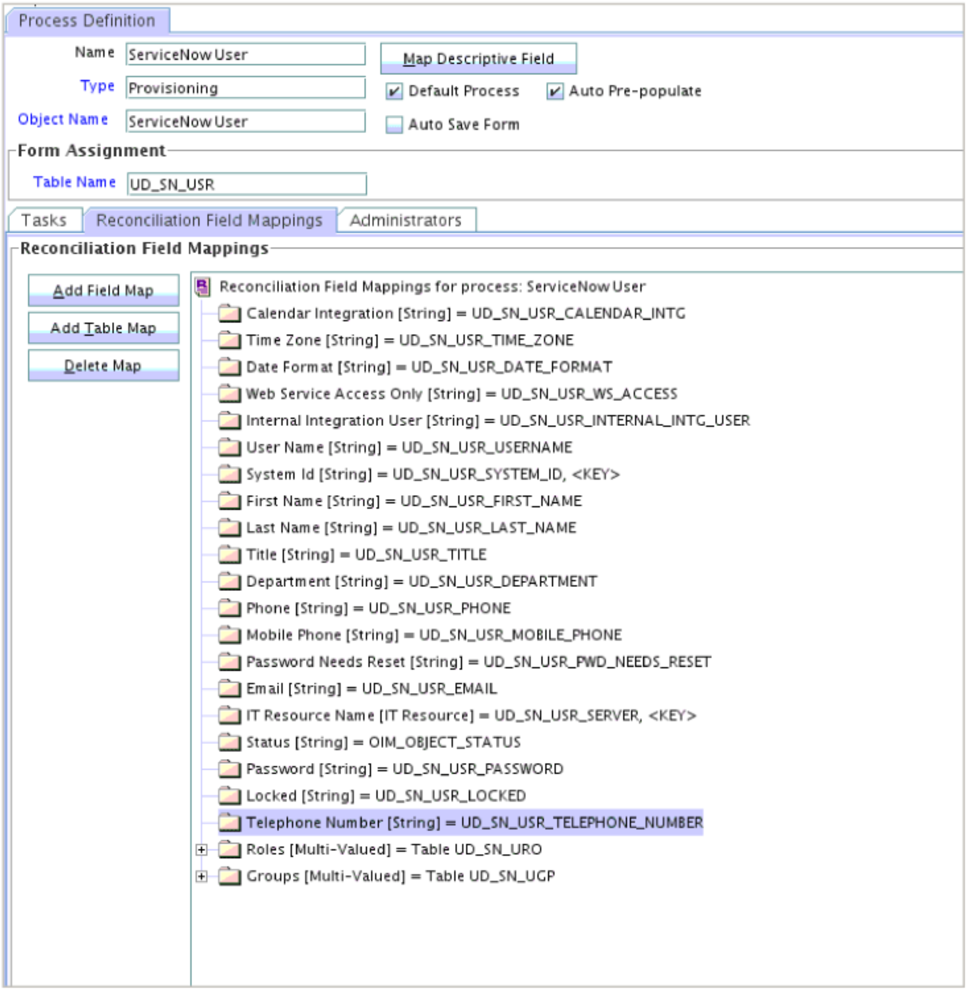
4.1.3 Creating Reconciliation Field Mapping

You create a reconciliation field mapping for the new attribute in the Process Definition section of Oracle Identity Manager Design Console.

1. Expand **Process Management**, and double-click **Process Definition**.
2. Search for and open the **ServiceNow User** process definition.
3. On the Reconciliation Field Mappings tab of the process definition, click **Add Field Map**.

4. From the Field Name list, select the field that you want to map.
5. Double-click the **Process Data Field** field, and then select the column for the attribute. For example, select **UD_TELEPHONENUMBER**.
6. Click the **Save** icon. The following screenshot shows the new reconciliation field mapped to a process data field in the process definition:

Figure 4-3 New Reconciliation Field Mapped to a Process Data Field in the Process Definition



Name: ServiceNow User
Type: Provisioning
Object Name: ServiceNow User

☑ Default Process ☑ Auto Pre-populate
☐ Auto Save Form

Form Assignment
Table Name: UD_SN_USR

Reconciliation Field Mappings for process: ServiceNow User
- Calendar Integration [String] = UD_SN_USR_CALENDAR_INTG
- Time Zone [String] = UD_SN_USR_TIME_ZONE
- Date Format [String] = UD_SN_USR_DATE_FORMAT
- Web Service Access Only [String] = UD_SN_USR_WS_ACCESS
- Internal Integration User [String] = UD_SN_USR_INTERNAL_INTG_USER
- User Name [String] = UD_SN_USR_USERNAME
- System Id [String] = UD_SN_USR_SYSTEM_ID, <KEY>
- First Name [String] = UD_SN_USR_FIRST_NAME
- Last Name [String] = UD_SN_USR_LAST_NAME
- Title [String] = UD_SN_USR_TITLE
- Department [String] = UD_SN_USR_DEPARTMENT
- Phone [String] = UD_SN_USR_PHONE
- Mobile Phone [String] = UD_SN_USR_MOBILE_PHONE
- Password Needs Reset [String] = UD_SN_USR_PWD_NEEDS_RESET
- Email [String] = UD_SN_USR_EMAIL
- IT Resource Name [IT Resource] = UD_SN_USR_SERVER, <KEY>
- Status [String] = OIM_OBJECT_STATUS
- Password [String] = UD_SN_USR_PASSWORD
- Locked [String] = UD_SN_USR_LOCKED
- Telephone Number [String] = UD_SN_USR_TELEPHONE_NUMBER
- Roles [Multi-Valued] = Table UD_SN_URO
- Groups [Multi-Valued] = Table UD_SN_UGP

4.1.4 Creating Entries in Lookup Definition for Reconciliation

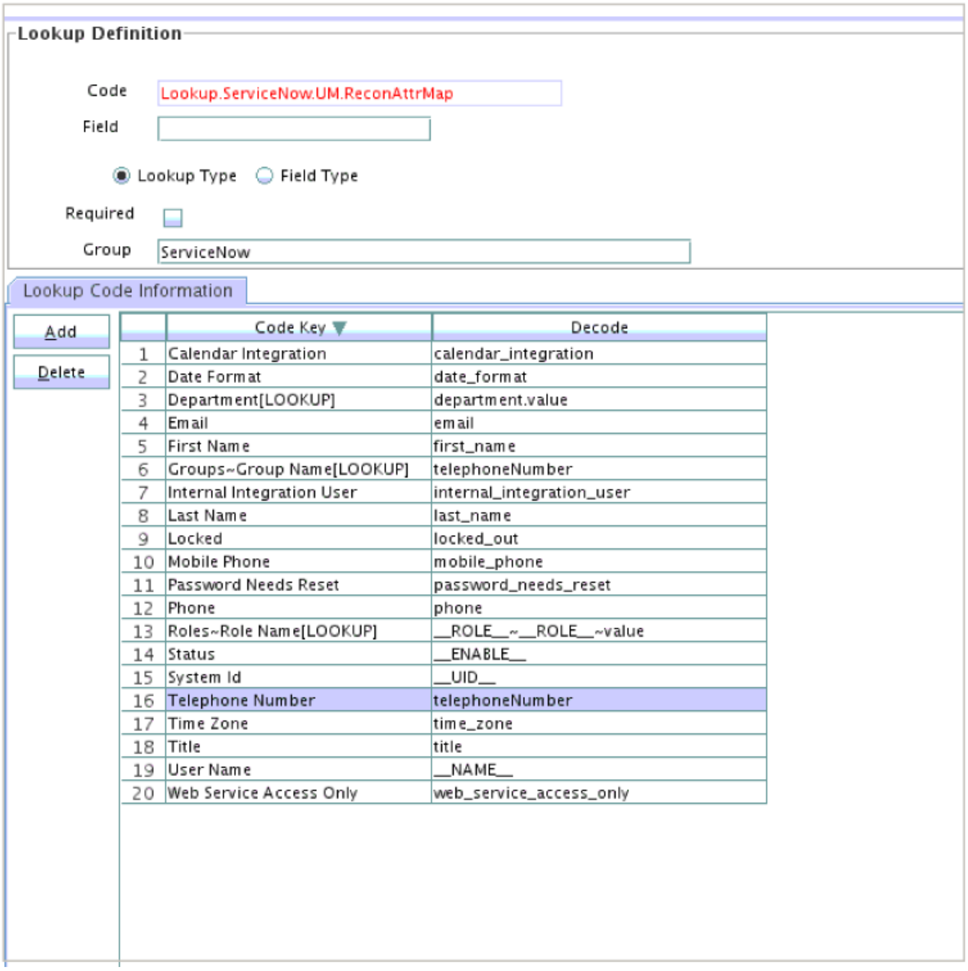
You create an entry for the newly added attribute in the lookup definition that holds attribute mappings for reconciliation.

1. Expand **Administration**.
2. Double-click **Lookup Definition**.
3. Search for and open the **Lookup.ServiceNow.UM.ReconAttrMap** lookup definition.
4. Click **Add** and enter the Code Key and Decode values for the field.

5. Click the Save icon.

   The following screenshot shows the entry added to the lookup definition:

Figure 4-4 Entry Added to the Lookup Definition



Lookup Definition

Code: Lookup.ServiceNow.UM.ReconAttrMap

Field:

● Lookup Type ○ Field Type

Required □

Group: ServiceNow

Lookup Code Information

	Code Key ▼	Decode
1	Calendar Integration	calendar_integration
2	Date Format	date_format
3	Department[LOOKUP]	department.value
4	Email	email
5	First Name	first_name
6	Groups~Group Name[LOOKUP]	telephoneNumber
7	Internal Integration User	internal_integration_user
8	Last Name	last_name
9	Locked	locked_out
10	Mobile Phone	mobile_phone
11	Password Needs Reset	password_needs_reset
12	Phone	phone
13	Roles~Role Name[LOOKUP]	__ROLE__~__ROLE__~value
14	Status	__ENABLE__
15	System Id	__UID__
16	Telephone Number	telephoneNumber
17	Time Zone	time_zone
18	Title	title
19	User Name	__NAME__
20	Web Service Access Only	web_service_access_only

4.1.5 Performing Changes in a New UI Form

You must replicate all changes made to the Form Designer of the Design Console in a new UI form.

1. Log in to Oracle Identity System Administration.
2. Create and activate a sandbox. See Creating and Activating a Sandbox.
3. Create a new UI form to view the newly added field along with the rest of the fields. See Creating a New UI Form.
4. Associate the newly created UI form with the application instance of your target system. To do so, open the existing application instance for your resource, from the Form field, select the form, and then save the application instance.
5. Publish the sandbox. See Publishing a Sandbox.

4.2 Adding New User Attributes for Provisioning

The connector provides a default set of attribute mappings for provisioning between Oracle Identity Manager and the target system. If required, you can add new user attributes for provisioning.

The default attribute mappings for provisioning are listed in Table 1-12.

The following topics discuss the procedure to add new user or group attributes for provisioning:

- Adding New Attributes for Provisioning
- Creating Entries in Lookup Definition for Provisioning
- Creating a Task to Enable Update Operations
- Replicating Form Designer Changes to a New UI Form

4.2.1 Adding New Attributes for Provisioning

You add a new attribute on the process form in the Form Designer section of Oracle Identity Manager Design Console.

> **Note:**
>
> If you have already added an attribute for reconciliation, then you need not repeat steps performed as part of that procedure.

1. Log in to the Oracle Identity Manager Design Console.
2. Expand **Development Tools**, and double-click **Form Designer**.
3. Search for and open the **UD_ServiceNow_USR** process form.
4. Click **Create New Version**, and then click **Add**.
5. Enter the details of the attribute.

   For example, if you are adding the TELEPHONENUMBER field, enter `UD_TELEPHONENUMBER` in the Name field, and then enter the rest of the details of this field.
6. Click the Save icon, and then click **Make Version Active**.

   The following screenshot shows the new field added to the process form:

Figure 4-5   New Field Added to the Process Form

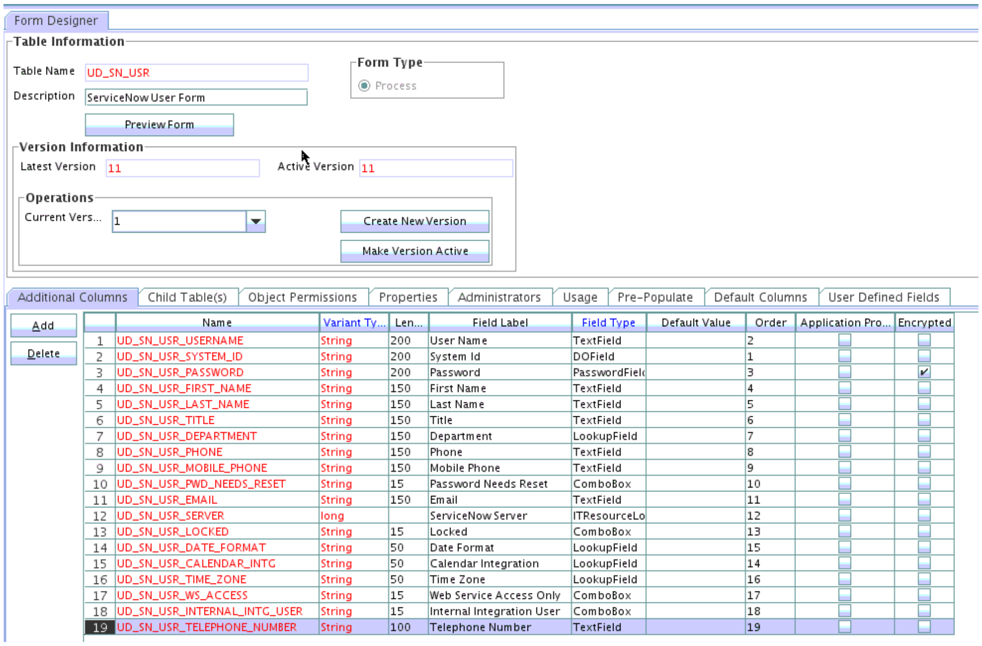


Table Information

Table Name: UD_SN_USR

Description: ServiceNow User Form

Form Type: ● Process

Version Information

Latest Version: 11

Active Version: 11

Operations

Current Vers...: 1

	Name	Variant Ty...	Len...	Field Label	Field Type	Default Value	Order	Application Pro...	Encrypted
1	UD_SN_USR_USERNAME	String	200	User Name	TextField		2	□	□
2	UD_SN_USR_SYSTEM_ID	String	200	System Id	DOField		1	□	□
3	UD_SN_USR_PASSWORD	String	200	Password	PasswordFiel(		3	□	☑
4	UD_SN_USR_FIRST_NAME	String	150	First Name	TextField		4	□	□
5	UD_SN_USR_LAST_NAME	String	150	Last Name	TextField		5	□	□
6	UD_SN_USR_TITLE	String	150	Title	TextField		6	□	□
7	UD_SN_USR_DEPARTMENT	String	150	Department	LookupField		7	□	□
8	UD_SN_USR_PHONE	String	150	Phone	TextField		8	□	□
9	UD_SN_USR_MOBILE_PHONE	String	150	Mobile Phone	TextField		9	□	□
10	UD_SN_USR_PWD_NEEDS_RESET	String	15	Password Needs Reset	ComboBox		10	□	□
11	UD_SN_USR_EMAIL	String	150	Email	TextField		11	□	□
12	UD_SN_USR_SERVER	long		ServiceNow Server	ITResourceLo		12	□	□
13	UD_SN_USR_LOCKED	String	15	Locked	ComboBox		13	□	□
14	UD_SN_USR_DATE_FORMAT	String	50	Date Format	LookupField		15	□	□
15	UD_SN_USR_CALENDAR_INTG	String	50	Calendar Integration	LookupField		14	□	□
16	UD_SN_USR_TIME_ZONE	String	50	Time Zone	LookupField		16	□	□
17	UD_SN_USR_WS_ACCESS	String	15	Web Service Access Only	ComboBox		17	□	□
18	UD_SN_USR_INTERNAL_INTG_USER	String	15	Internal Integration User	ComboBox		18	□	□
19	UD_SN_USR_TELEPHONE_NUMBER	String	100	Telephone Number	TextField		19	□	□

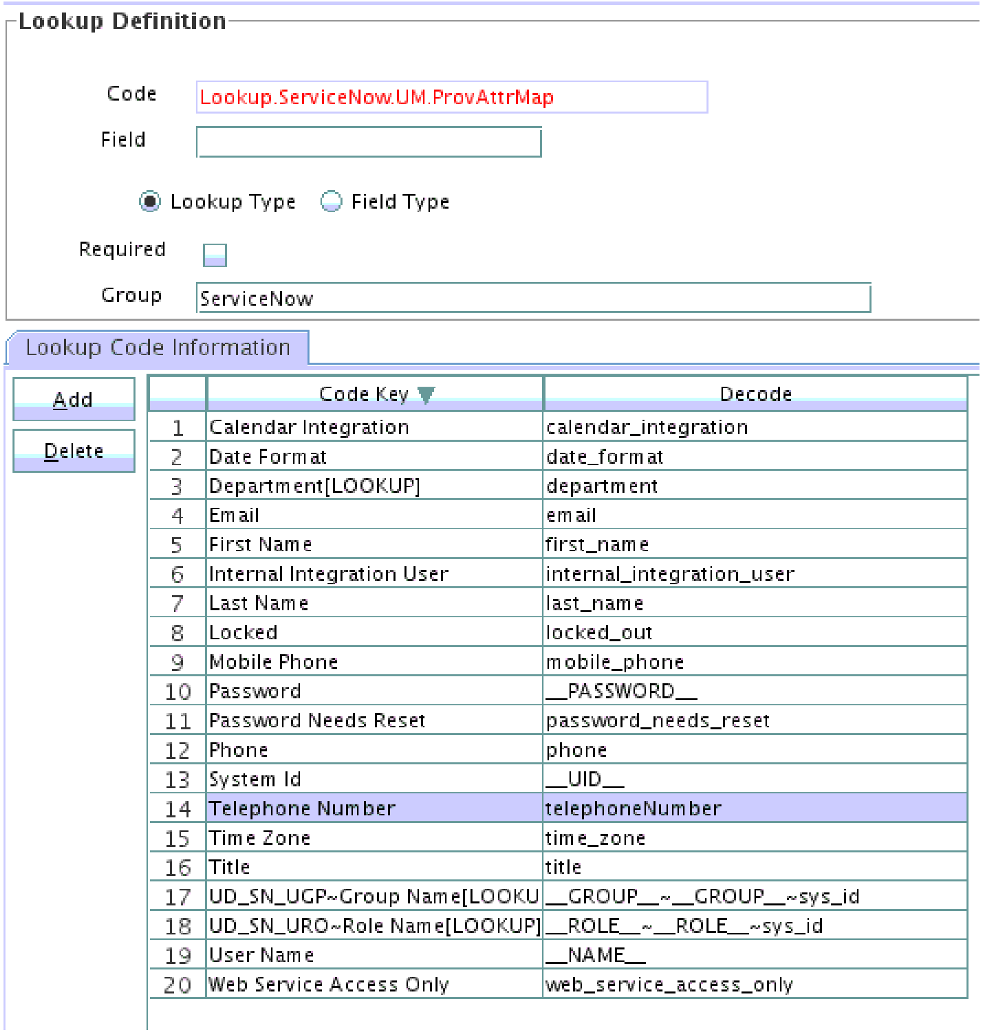
4.2.2 Creating Entries in Lookup Definition for Provisioning

You create an entry for the newly added attribute in the lookup definition that holds attribute mappings for provisioning.

1. Expand **Administration**.
2. Double-click **Lookup Definition**.
3. Search for and open the **Lookup.ServiceNow.UM.ProvAttrMap** lookup definition.
4. Click **Add** and enter the Code Key and Decode values for the field.
5. Click the Save icon.

   The following screenshot shows the entry added to the lookup definition:

Figure 4-6 Entry Added to the Lookup Definition



Code: Lookup.ServiceNow.UM.ProvAttrMap

Field:

● Lookup Type ○ Field Type

Required □

Group: ServiceNow

	Code Key	Decode
1	Calendar Integration	calendar_integration
2	Date Format	date_format
3	Department[LOOKUP]	department
4	Email	email
5	First Name	first_name
6	Internal Integration User	internal_integration_user
7	Last Name	last_name
8	Locked	locked_out
9	Mobile Phone	mobile_phone
10	Password	__PASSWORD__
11	Password Needs Reset	password_needs_reset
12	Phone	phone
13	System Id	__UID__
14	Telephone Number	telephoneNumber
15	Time Zone	time_zone
16	Title	title
17	UD_SN_UGP~Group Name[LOOKU	__GROUP__~__GROUP__~sys_id
18	UD_SN_URO~Role Name[LOOKUP]	__ROLE__~__ROLE__~sys_id
19	User Name	__NAME__
20	Web Service Access Only	web_service_access_only

4.2.3 Creating a Task to Enable Update Operations

Create a task to enable updates on the new user or group attribute during provisioning operations. The connector provides a default set of attribute mappings for provisioning between Oracle Identity Manager and the target system. If required, you can add new user or group attributes for provisioning.

To enable the update of the attribute during provisioning operations, add a process task for updating the new user attribute as follows:

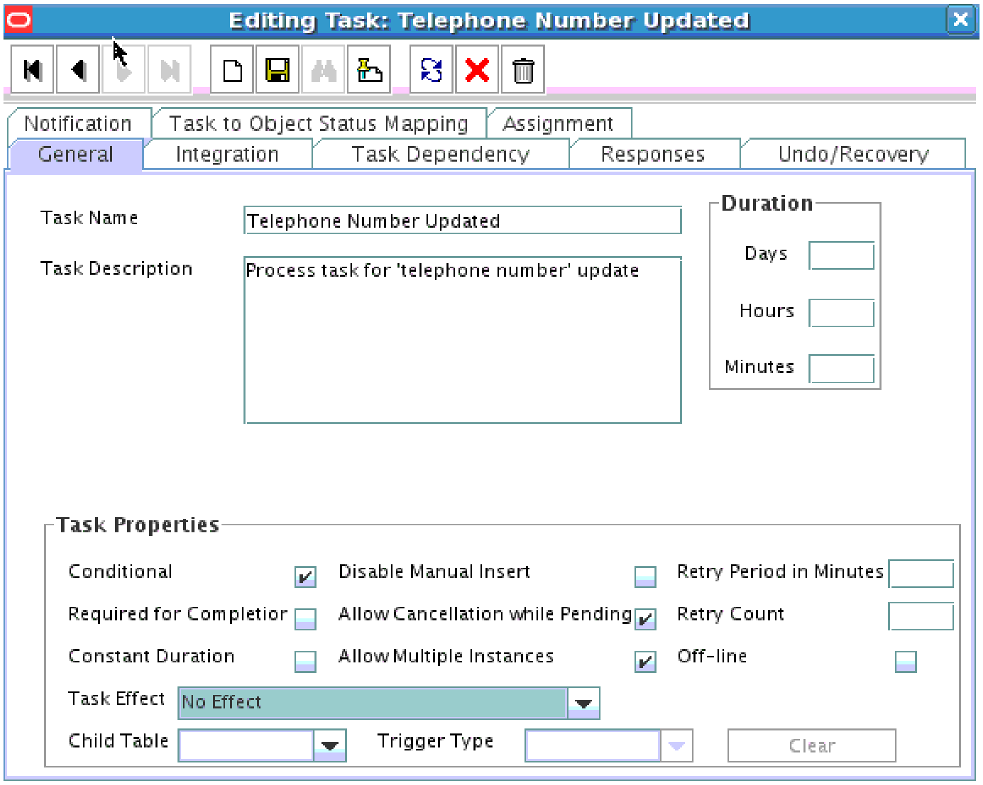
1. Expand **Process Management**, and double-click **Process Definition**.
2. Search for and open the **ServiceNow User** process definition.
3. Click **Add.**
4. On the General tab of the Creating New Task dialog box, enter a name and description for the task and then select the following:
   - Conditional
   - Allow Cancellation while Pending

- Allow Multiple Instances

5. Click the Save icon.

   The following screenshot shows the new task added to the process definition:

Figure 4-7   New task Added to the Process Definition



Task Name	Telephone Number Updated
Task Description	Process task for 'telephone number' update

Duration

Days	
Hours	
Minutes	

Task Properties

Conditional	☑	Disable Manual Insert	□	Retry Period in Minutes	
Required for Completior	□	Allow Cancellation while Pending	☑	Retry Count	
Constant Duration	□	Allow Multiple Instances	☑	Off-line	□
Task Effect	No Effect				
Child Table		Trigger Type			

6. In the provisioning process, select the adapter name in the Handler Type section as follows:

   a. Go to the Integration tab, click **Add.**

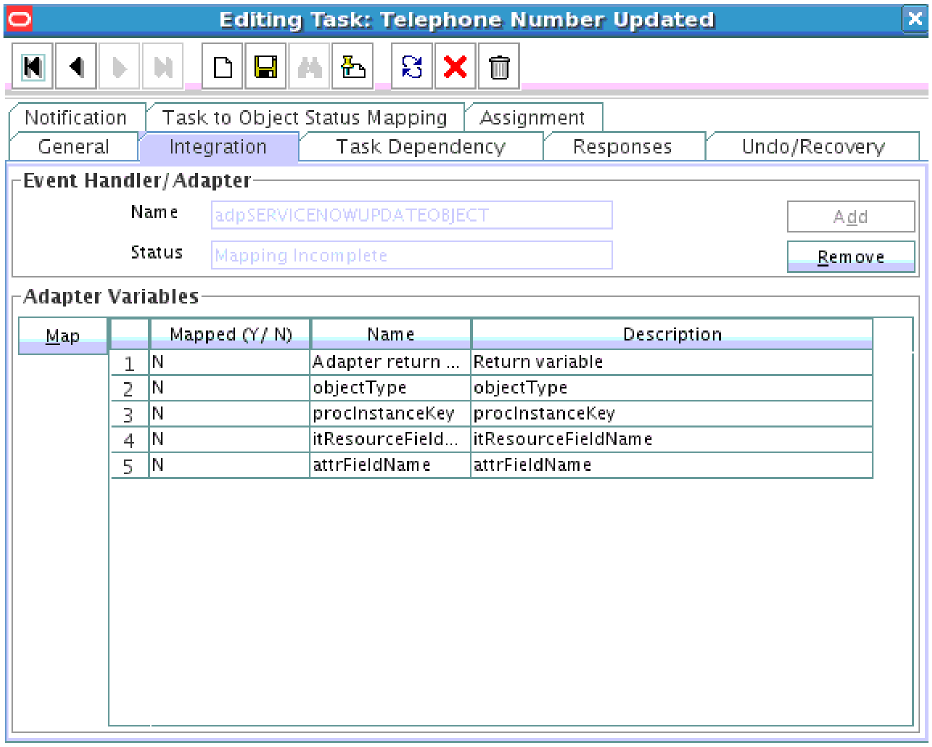
   b. In the Handler Selection dialog box, select **Adapter**.

   c. From the Handler Name column, select **adpSERVICENOWUPDATEOBJECT**.

   d. Click Save and close the dialog box.

      The list of adapter variables is displayed on the Integration tab. The following screenshot shows the list of adapter variables:

Figure 4-8 List of Adapter Variables



Name: adpSERVICENOWUPDATEOBJECT

Status: Mapping Incomplete

	Mapped (Y/ N)	Name	Description
1	N	Adapter return ...	Return variable
2	N	objectType	objectType
3	N	procInstanceKey	procInstanceKey
4	N	itResourceField...	itResourceFieldName
5	N	attrFieldName	attrFieldName

7. In the Adapter Variables region, click the **ParentFormProcessInstanceKey** variable.
8. In the dialog box that is displayed, create the following mapping:
   - **Variable Name:** ParentFormProcessInstanceKey
   - **Map To:** Process Data
   - **Qualifier:**Process Instance
9. Click Save and close the dialog box.
10. If you are enabling update provisioning operations for a User attribute, then repeat Steps 7 through 9 for the remaining variables listed in the Adapter Variables region.

    The following table lists values that you must select from the Map To, Qualifier, and Literal Value lists for each variable:

Variable	Map To	Qualifier	Literal Value
Adapter Return Value	Response Code	NA	NA
Object Type	Literal	String	User
itResourceFieldName	Literal	String	UD_SN_USR_SERVER
attributeFieldName	Literal	String	Telephone Number

11. On the Responses tab, click Add to add at least the SUCCESS response code, with Status C. This ensures that if the task is successfully run, then the status of the task is displayed as `Completed`.

12. Click the Save icon and close the dialog box, and then save the process definition.

4.2.4 Replicating Form Designer Changes to a New UI Form

You must replicate all changes made to the Form Designer of the Design Console in a new UI form.

1. Log in to Oracle Identity System Administration.
2. Create and activate a sandbox. See Creating and Activating a Sandbox.
3. Create a new UI form to view the newly added field along with the rest of the fields. See Creating a New UI Form.
4. Associate the newly created UI form with the application instance of your target system. To do so, open the existing application instance for your resource, from the Form field, select the form, and then save the application instance.
5. Publish the sandbox. See Publishing a Sandbox.

4.3 Configuring Validation of Data During Reconciliation and Provisioning

You can configure validation of reconciled and provisioned single-valued data according to your requirements. For example, you can validate data fetched from the User Name attribute to ensure that it does not contain the number sign (#). In addition, you can validate data entered in the User Name field on the process form so that the number sign (#) is not sent to the target system during provisioning operations. For data that fails the validation check, the following message is displayed or recorded in the log file: Validation failed for attribute ATTRIBUTE_NAME.

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

To configure validation of data:

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

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

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

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

   ```
   public boolean validate(HashMap hmUserDetails,
   HashMap hmEntitlementDetails, String field) { /*
   *    You must write code to validate attributes. Parent
   ```

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

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

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

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

   For Microsoft Windows: *OIM_HOME/server/bin/UploadJars.bat*

   For UNIX: *OIM_HOME/server/bin/UploadJars.sh*

   When you run the utility, you are prompted to enter the login credentials of the Oracle Identity Manager administrator, URL of the Oracle Identity Manager host computer, context factory value, type of JAR file being uploaded, and the location from which the JAR file is to be uploaded. Specify 1 as the value of the JAR type.
4. If you created the Java class for validating a process form field for reconciliation, then:
   a. Log in to the Design Console.
   b. Create a lookup definition named **Lookup.ServiceNow.UM.ReconValidation**.
   c. In the Code Key column, enter the resource object field
      name that you want to validate. For example, `Firstname`.
      In the Decode column, enter the class name For example,
      `org.identityconnectors.servicenow.extension.servicenowAMValidator`.
   d. Save the changes to the lookup definition.

   e. Search for and open the LookupServiceNow.UM.Configuration lookup definition.
   f. In the Code Key column, enter `Recon Validation Lookup`. In the Decode column, enter `Lookup.ServiceNow.UM.ReconValidation`.
   g. Save the changes to the lookup definition.

5. If you created the Java class for validating a process form field for provisioning, then:
   a. Log in to the Design Console.
   b. Create a lookup definition named **Lookup.ServiceNow.UM.ProvValidation**.
   c. In the Code Key column, enter.the process form field label. For example, `Firstname`. In the Decode column, enter the class name. For example, `org.identityconnectors.ServiceNow.extension.ServiceNowValidator`.
   d. Save the changes to the lookup definition.
   e. Search for and open the `Lookup.ServiceNow.UM.Configuration` lookup definition.
   f. In the Code Key column, enter `Provisioning Validation Lookup`. In the Decode column, enter `Lookup.ServiceNow.UM.ProvValidation`.
   g. Save the changes to the lookup definition.
6. Purge the cache to get the changes reflected in Oracle Identity Manager.
   See Purging Cache in *Oracle Fusion Middleware Administering Oracle Identity Manager* for information about purging cache.

4.4 Configuring Transformation of Data During User Reconciliation

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

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

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

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

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

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

```
package oracle.iam.connectors.common.transform;
import java.util.HashMap;
public class TransformAttribute {
/*
Description:Abstract method for transforming the attributes
param hmUserDetails< String,Object>
HashMap containing parent data details
param hmEntitlementDetails < String,Object>
HashMap containing child data details
*/
public Object transform(HashMap hmUserDetails, HashMap
hmEntitlementDetails,String sField) {
/*
*    You must write code to transform the attributes. Parent data
attribute values can be fetched by using hmUserDetails.get("Field
Name").
*To fetch child data values, loop through the
*    ArrayList/Vector fetched by hmEntitlementDetails.get("Child
Table")
*    Return the transformed attribute.
*/
String sUserName= (String)hmUserDetails.get("User Name");
String sLastName= (String)hmUserDetails.get("Last Name"); String
sFullName=sUserName+"."+sLastName;
return sFullName;
}
}
```

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

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

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

   - **For Microsoft Windows**: *OIM_HOME/server/bin/UploadJars.bat*
   - **For UNIX**: *OIM_HOME/server/bin/UploadJars.sh*

   When you run the utility, you are prompted to enter the login credentials of the Oracle Identity Manager administrator, URL of the Oracle Identity Manager host computer, context factory value, type of JAR file being uploaded, and the location from which the JAR file is to be uploaded. Specify 1 as the value of the JAR type.
4. Create a new lookup definition for transformation as follows:
   a. Log in to the Design Console.
   b. Expand Administration, and then double-click Lookup Definition.
   c. In the Code field, enter `Lookup.ServiceNow.UM.ReconTransformations` as the name of the lookup definition.

   d. Select the **Lookup Type** option.

   e. On the **Lookup Code Information** tab, click **Add**.

   f. In the Code Key column, enter the resource object field name on which you want to apply transformation. For example, User Name. In the Decode column, enter the name of the class that implements the transformation logic. For example, oracle.iam.connectors.common.transform.TransformAttribute.

   g. Save the changes to the lookup definition.

5. Add an entry in the **Lookup.ServiceNow.UM.Configuration** lookup definition to enable transformation as follows:

   a. Expand Administration, and then double-click **Lookup Definition**.

   b. Search for and open the **Lookup.ServiceNow.UM.Configuration** lookup definition.

   c. In the Code Key column, enter Recon Transformation Lookup. In the Decode column, enter Lookup.ServiceNow.UM.ReconTransformation.

   d. Save the changes to the lookup definition.

6. Purge the cache to get the changes reflected in Oracle Identity Manager. See Purging Cache in *Oracle Fusion Middleware Administering Oracle Identity Manager* for information about purging cache.

4.5 Configuring the Connector for Multiple Installations of the Target System

You might want to configure the connector for multiple installations of the target system.

The following example illustrates this requirement:

The London and New York offices of Example Multinational Inc. have their own installations of the target system. The company has recently installed Oracle Identity Manager, and they want to configure Oracle Identity Manager to link all the installations of the target system.

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

4.6 Defining the Connector

By using the Oracle Identity System Administration, 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 Known Issues and Workarounds for the ServiceNow Connector

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

A

Files and Directories of the ServiceNow Connector

These are the components of the connector installation media that comprise the connector.

Table A-1   Files and Directories On the Installation Media

File in the Installation Media Directory	Description
/bundle/ org.identityconnectors.servicenow-1.0.1115.jar	This JAR file is the ICF connector bundle.
configuration/ServiceNow-CI.xml	This XML file contains configuration information that is used 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 the Oracle Identity Manager database. **Note:** A resource bundle is a file containing localized versions of the text strings that include GUI element labels and messages.
xml/ServiceNow-ConnectorConfig.xml	This XML file contains definitions for the following connector objects: • IT resource definition • Process forms • Process tasks and adapters • Lookup definitions • Resource objects • Process definition • Scheduled tasks • Reconciliation rules

Index