Oracle® Identity Manager

Connector Guide for GoToMeeting

Release 11.1.1
E78206-07
July 2020

ORACLE

Oracle Identity Manager Connector Guide for GoToMeeting, Release 11.1.1

E78206-07

Primary Author: Gowri GR

Contributing Authors: Alankrita Prakash

Contributors: Neha Bagalkot

Contents

A Files and Directories on the GoToMeeting Connector Installation Media

List of Figures

List of Tables

Preface

This guide describes the connector that is used to integrate Oracle Identity Manager with GoToMeeting. You can implement the GoToMeeting connector either by using GoToMeeting with Oracle Identity Manager Connector for Microsoft Active Directory User Management or by using the predefined integration.

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 the Oracle Identity Manager Connector for Microsoft Active Directory User Management, visit the following Oracle Help Center page:

https://docs.oracle.com/cd/E22999_01/doc.111/e20347/toc.htm

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

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

Conventions

The following text conventions are used in this document:

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

What's New in Oracle Identity Manager Connector for GoToMeeting?

These are the updates made to the software and documentation for the Oracle Identity Manager Connector for GoToMeeting (GoToMeeting connector) in release 11.1.1.5.0.

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

- Software Updates

  This section provides information on the updates that are made to the connector software. This section also provides information on the sections of this guide that have changed in response to each software update.

- Documentation-Specific Updates

  This section provides information on the major changes that are 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

These are the updates made to the connector software.

Software Updates in Release 11.1.1.5.0

This is the first release of the GoToMeeting connector. Therefore, there are no software updates in this release.

Documentation-Specific Updates

These are the updates made to the connector documentation.

Documentation-Specific Updates in Release 11.1.1.5.0

The following is a documentation-specific update in revision "07" of this guide:

The "Oracle Identity Governance or Oracle Identity Manager" and "Target systems" row of Table 2-1 and Table 3-1 have been updated.

The following is a documentation-specific update in revision "06" of this guide:

The "Oracle Identity Governance or Oracle Identity Manager" row of Table 2-1 and Table 3-1 have been updated to include support for Oracle Identity Governance release 12*c* PS4 (12.2.1.4.0).

The following is a documentation-specific update in revision "05" of this guide:

The "Oracle Identity Manger" row of Table 2-1 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 is a documentation-specific update in revision "04" of this guide:

References to “Citrix” have been removed from the guide due to rebranding of the target system.

The following is a documentation-specific update in revision "03" of this guide:

The Enabling Logging section has been modified to update the logger name from "ORG.IDENTITYCONNECTORS.GENERICREST" to "ORG.IDENTITYCONNECTORS.GoToMeeting".

The following is a documentation-specific update in revision "02" of this guide:

The “host” and “Access Token Endpoint” rows of Table 3-11 and Table 3-15 have been updated to include the latest host name of the target system.

The following is a documentation-specific update in revision “01” of this guide:

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

1

Overview of the GoToMeeting Connector

Oracle Identity Manager is a centralized identity management solution that provides self service, compliance, provisioning, and password management services for applications residing on-premise or on the Cloud. Oracle Identity Manager connectors are used to integrate Oracle Identity Manager with the external and identity-aware applications. This connector integrates Oracle Identity Manager with GoToMeeting. This chapter contains the following sections:

- Introduction to the GoToMeeting Connector
- Use Cases Supported by the GoToMeeting Connector
- Roadmap for Implementing the GoToMeeting Connector

1.1 Introduction to the GoToMeeting Connector

The GoToMeeting connector enables Oracle Identity Manager to manage the identity data for GoToMeeting by integrating Oracle Identity Manager with the GoToMeeting Admin Center (Admin Center) either by using Oracle Identity Manager Connector for Microsoft Active Directory User Management (OIM AD connector) or by using the predefined GoToMeeting connector.

Both the implementations can be configured to run in the account management (or target resource management) mode.

The following sections provide details on the GoToMeeting connector implementations:

- OIM AD Connector Integration
- Predefined Integration

OIM AD Connector Integration

This implementation of the connector integrates Oracle Identity Manager with the Admin Center by using Microsoft Active Directory (AD) as a middleware. In this implementation, AD is used as a user source for performing user management operations in the Admin Center.

The OIM AD connector allows synchronization of the GoToMeeting user information between Oracle Identity Manager and AD. The GoToMeeting Active Directory Connector (GoToMeeting AD connector) helps in fetching the attributes from AD and then synchronizing the data with the Admin Center.

This is an optional integration, and you can choose this integration for managing users if you have already configured Microsoft Active Directory with the Admin Center.

See Integrating GoToMeeting with OIM AD Connector.

Predefined Integration

You can implement the GoToMeeting connector using the predefined integration. This implementation of the connector integrates Oracle Identity Manager with the Admin

Center by using GoToMeeting Administration APIs. The Admin Center is used as a managed (target) resource of the identity data for Oracle Identity Manager.

The information about users that are created or modified directly on the Admin Center can be reconciled into Oracle Identity Manager. This data is used to add or modify resources (that is, accounts) that are allocated to Oracle Identity Manager Users. In addition, you can use Oracle Identity Manager to provision or update GoToMeeting accounts that are assigned to Oracle Identity Manager Users.

See Implementing the GoToMeeting Connector.

> **Note:**
>
> In this guide:
>
> - Oracle Identity Manager Connector for Microsoft Active Directory User Management is referred to as the **OIM AD connector**.
> - GoToMeeting Active Directory Connector is referred to as the **GoToMeeting AD connector**.
> - GoToMeeting Admin Center is referred to as the **target system**.

1.2 Use Cases Supported by the GoToMeeting Connector

GoToMeeting is a cloud-based application that offers online meeting, screen sharing, and video conferencing features. The GoToMeeting connector helps in managing GoToMeeting users and their accounts through Oracle Identity Manager.

The following is a common scenario in which the GoToMeeting connector can be used:

Organizations use GoToMeeting for a real-time collaboration of users across various locations. The administrator needs to assign an account with a valid GoToMeeting license to each GoToMeeting user, and also ensure that the user is unable to access the application using the account after leaving the organization.

The GoToMeeting connector provides a user management functionality that enables automation of provisioning and deprovisioning of GoToMeeting user accounts. The connector enables organizations to manage identities and licenses for GoToMeeting users through Oracle Identity Manager. For example, after a user joins an organization, an account is automatically provisioned to the user based on the predefined access policies in Oracle Identity Manager. In addition, the user is assigned a GoToMeeting license. Similarly, this account is deactivated after the user leaves the organization. This saves time and provides robust security because of less manual intervention.

1.3 Roadmap for Implementing the GoToMeeting Connector

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

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

- Integrating GoToMeeting with OIM AD Connector provides information on understanding the OIM AD connector integration with GoToMeeting. It also

provides guidelines on deploying and using the required connectors (OIM AD connector and GoToMeeting AD connector) for implementing this integration approach.

- Implementing the GoToMeeting Connector provides information on understanding, deploying, and using the predefined GoToMeeting connector.
- Files and Directories on the GoToMeeting Connector Installation Media lists the files and directories that comprise the connector installation media.

2

Integrating GoToMeeting with OIM AD Connector

In this integration, the user management operations are implemented in the GoToMeeting Admin Center (Admin Center) by using Microsoft Active Directory as a middleware. The OIM AD connector and GoToMeeting AD connector help in synchronizing user attributes between Oracle Identity Manager, Microsoft Active Directory, and GoToMeeting directory services.

This chapter contains the following sections:

All other information on this connector (such as certified languages, supported features, lookup definitions used during connector operations, and so on) is available in *Oracle Identity Manager Connector Guide for Microsoft Active Directory User Management*.

2.1 Certified Components for the OIM AD Connector Integration

These are the software components and their versions required for integrating Oracle Identity Manager with GoToMeeting using the OIM AD connector.

Table 2-1 Certified Components for the OIM AD Connector Integration

Component	Requirement
Oracle Identity Governance or Oracle Identity Manager	You can use one of the following releases of Oracle Identity Manager or Oracle Identity Governance: • Oracle Identity Governance 12*c* (12.2.1.4.0) • Oracle Identity Governance 12*c* (12.2.1.3.0) • Oracle Identity Manager 11*g* Release 2 PS3 BP06 (11.1.2.3.6)
Target system	LogMeIn Admin Center
Oracle Identity Manager Connector for Microsoft Active Directory User Management	11.1.1.6.0
GoToMeeting Active Directory Connector	1.5.1.68

2.2 Architecture of the OIM AD Connector Integration

In this implementation, Microsoft Active Directory is used as a managed (target) resource of the identity data for Oracle Identity Manager.

The following connectors are used to synchronize data between Oracle Identity Manager and the Admin Center:

- **Oracle Identity Manager Connector for Microsoft Active Directory User Management**

  The OIM AD connector allows synchronization of user information between Oracle Identity Manager and AD, and is configured to run in the account management mode. This mode enables the following operations:

  - **Provisioning**

    Provisioning involves creating, updating, or deleting users on AD through Oracle Identity Manager. When you allocate (or provision) a Microsoft Active Directory resource to an Oracle Identity Manager User, the operation results in the creation of an account on Microsoft Active Directory for that user. In the Oracle Identity Manager context, the term **provisioning** is also used to mean updates made to the AD account through Oracle Identity Manager.

  - **Target resource reconciliation**

    In target resource reconciliation, data related to newly created and modified accounts on AD can be reconciled and linked with existing Oracle Identity Manager Users and provisioned resources. To perform target resource reconciliation, the Active Directory User Target Reconciliation scheduled job is used.

  See Connector Architecture in *Oracle Identity Manager Connector Guide for Microsoft Active Directory User Management*.

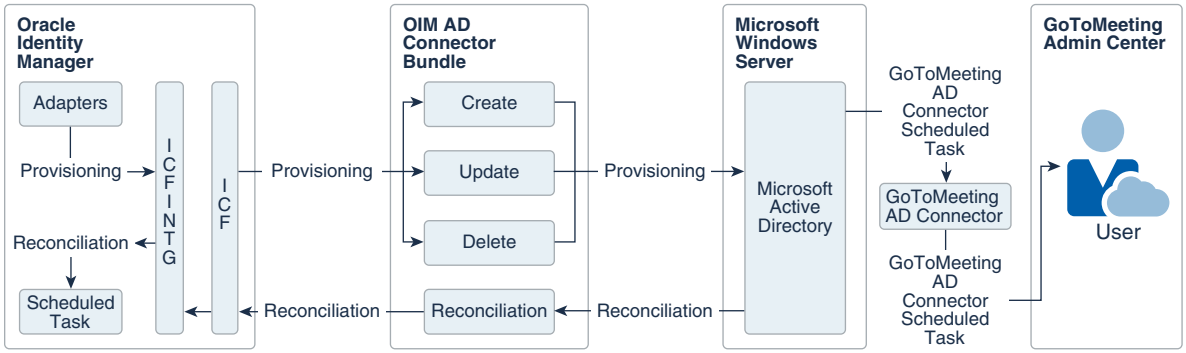
- **GoToMeeting Active Directory Connector**

  GoToMeeting uses a lightweight AD connector behind the firewall to synchronize user information between AD and GoToMeeting directory services. The GoToMeeting AD connector automates provisioning of user accounts to the Admin Center from AD. These user accounts are included as members of a Microsoft Active Directory group (specified as values of the AD Sync Group attribute of AD), which is used for synchronizing the accounts from AD to the Admin Center through a scheduled task.

  For details on the GoToMeeting AD connector, visit the GoToMeeting website at https://www.gotomeeting.com/, navigate to Support, and search for Active Directory Connector.

Figure 2-1 depicts the components used for integrating Oracle Identity Manager with GoToMeeting using the OIM AD connector.

Figure 2-1 Architecture of the OIM AD Connector Integration



As shown in Figure 2-1, AD is configured as a target resource of Oracle Identity Manager. The OIM AD connector is a .NET framework-based connector that is implemented using the Identity Connector Framework (ICF) component. The ICF component 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 connection pooling, buffering, time outs, and filtering. ICF is distributed together with Oracle Identity Manager. Therefore, you do not need to configure or modify ICF.

This implementation enables provisioning of user accounts on the target system through the following two-step process:

1. The OIM AD connector creates or updates user accounts in AD through the provisioning operations that are performed on Oracle Identity Manager.
2. The GoToMeeting AD connector automates provisioning of the user accounts by fetching the attributes from AD and then synchronizing the data with the Admin Center through the GoToMeeting AD connector scheduled task. Based on the data fetched from AD, the user accounts are automatically created or updated in the Admin Center.

Through reconciliation, account data that is created and updated directly on AD is fetched into Oracle Identity Manager and stored against the corresponding Oracle Identity Manager Users.

2.3 Operations Supported by the OIM AD Connector Integration

These are the user management operations supported by integrating GoToMeeting with Oracle Identity Manager using the OIM AD connector.

This integration supports the following operations:

- Create User
- Update User
- Delete User
- Enable User
- Disable User

2.4 Deploying and Using the OIM AD Connector and GoToMeeting AD Connector

As a prerequisite for Oracle Identity Manager to communicate with Microsoft Active Directory and GoToMeeting, the OIM AD connector and GoToMeeting AD connector must be deployed and configured at the back end.

The detailed instructions for deploying and using the OIM AD connector is available in *Oracle Identity Manager Connector Guide for Microsoft Active Directory User Management*. For more information, see the following sections of the guide:

- Deploying the Connector for performing the preinstallation, installation, and postinstallation tasks
- Using the Connector for understanding the guidelines on using the connector, performing the connector operations, and uninstalling the connector
- Extending the Functionality of the Connector for extending the functionality of the connector to address your specific requirements

The procedure for deploying and using the GoToMeeting AD connector is available in the GoToMeeting product documentation. For the detailed instructions on deploying and using the GoToMeeting AD connector, visit the GoToMeeting website at https://www.gotomeeting.com/, navigate to Support, and search for Active Directory Connector.

3
Implementing the GoToMeeting Connector

You can implement the GoToMeeting connector using the out-of-the-box integration, which enables Oracle Identity Manager to integrate with GoToMeeting using GoToMeeting Administration APIs.

The following sections provide an overview of the connector, instructions on deploying and using the connector, and extending its functionality to address your specific requirements:

3.1 About the GoToMeeting Connector

This connector can be configured to run in the account management (or target resource management) mode. In this mode, the GoToMeeting Admin Center (Admin Center) is used as a managed (target) resource of the identity data for Oracle Identity Manager.

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

3.1.1 Certified Components for the GoToMeeting Connector

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

Table 3-1 Certified Components for the GoToMeeting Connector

Component	Requirement
Oracle Identity Governance or Oracle Identity Manager	You can use one of the following releases of Oracle Identity Manager or Oracle Identity Governance: • Oracle Identity Governance 12*c* (12.2.1.4.0) • Oracle Identity Governance 12*c* (12.2.1.3.0) • Oracle Identity Manager 11*g* Release 2 PS3 BP06 (11.1.2.3.6)
Target system	LogMeIn Admin Center
Connector Server	11.1.2.1.0
Connector Server JDK	JDK 1.6 or Later

3.1.2 Certified Languages for the GoToMeeting 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

3.1.3 Architecture of the GoToMeeting Connector

The connector uses GoToMeeting Administration APIs to synchronize user attributes between Oracle Identity Manager and GoToMeeting directory services, and is implemented using the Identity Connector Framework (ICF) component.

The connector enables the following operations:

- Provisioning

  Provisioning involves creating and updating users on the Admin Center through Oracle Identity Manager. When you allocate (or provision) a GoToMeeting resource to an Oracle Identity Manager User, the operation results in the creation of an account in the Admin Center for that user. In the Oracle Identity Manager context, the term **provisioning** is also used to mean updates made to the account through Oracle Identity Manager.

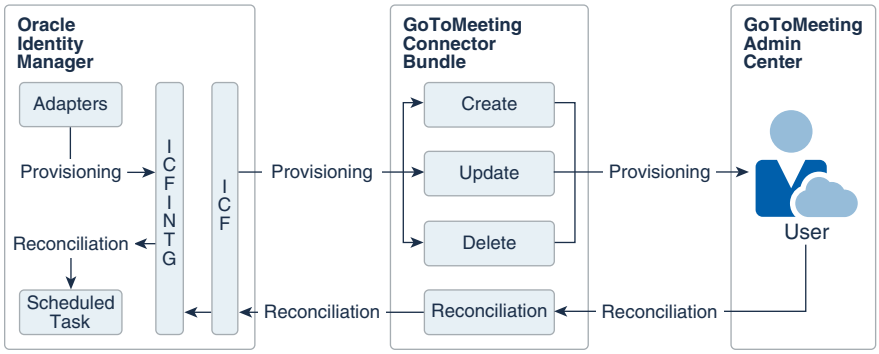
- Target resource reconciliation

  To perform target resource reconciliation, the GoToMeeting Reconciliation scheduled job is used. The connector then fetches the user attribute values from the Admin Center.

The connector supports OAuth 2.0 security protocol for authenticating to the target system, and uses access token and refresh token values as inputs from the user.

Figure 3-1 depicts the architecture of the GoToMeeting connector.

Figure 3-1 Architecture of the GoToMeeting Connector



As shown in Figure 3-1, the Admin Center is configured as a target resource of Oracle Identity Manager. Through the provisioning operations that are performed on Oracle Identity Manager, accounts are created and updated on the Admin Center for Oracle Identity Manager Users.

Through reconciliation, account data that is created and updated directly on the Admin Center is fetched into Oracle Identity Manager and stored against the corresponding Oracle Identity Manager Users.

The ICF component 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 connection pooling, buffering, time outs, and filtering. ICF is distributed together with Oracle Identity Manager. Therefore, you do not need to configure or modify ICF.

During provisioning, the adapters invoke ICF operation, ICF invokes the Create operation on GoToMeeting Connector Bundle, and then the bundle calls the OAuth API for connecting to the Admin Center and performing user authentication. The Admin Center accepts provisioning data from the bundle using GoToMeeting Administration APIs, carries out the operation, and returns the response back to the bundle. The bundle then passes it to the adapters.

3.1.4 Features of the GoToMeeting Connector

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

This connector supports the following features:

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

3.1.4.1 Full Reconciliation

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

3.1.4.2 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.

See Understanding Installation of the GoToMeeting 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, configuring, and running the connector server

3.1.4.3 Limited Reconciliation

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

See Performing Limited Reconciliation.

3.1.4.4 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

3.1.5 Lookup Definitions Used During Connector Operations

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

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

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

3.1.5.1 Preconfigured Lookup Definitions

Preconfigured lookup definitions are automatically created in Oracle Identity Manager after 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.GTM.Configuration
- Lookup.GTM.UM.Configuration
- Lookup.GTM.UM.ProvAttrMap
- Lookup.GTM.UM.ReconAttrMap
- Lookup.GTM.Locale

3.1.5.1.1 Lookup.GTM.Configuration

The Lookup.GTM.Configuration lookup definition holds connector configuration entries that are used during the target resource reconciliation and provisioning operations. Table 3-2 lists the default entries in this lookup definition.

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

Table 3-2 Entries in the Lookup.GTM.Configuration Lookup Definition

Code Key	Decode	Description
Bundle Name	org.identityconnectors.genericrest	This entry holds the name of the connector bundle.
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__.CREATEOP={\"users\":[{\"email\":\"$(email)$\",\"firstName\":\"$(__NAME__)$\",\"lastName\":\"$(lastName)$\",\"locale\":\"$(locale)$\"}],\"groupKey\":\"$(groupKey)$\",\"licenseKeys\":[$(licenseKeys)$]}"	This entry lists the request payload formats for all the connector operations that are not in standard JSON format.
httpHeaderAccept	application/json	This entry holds the accept type expected from the target system in the request header.
httpHeaderContentType	application/json	This entry holds the content type expected by the target system in the request header.
jsonResourcesTag	"__ACCOUNT__=results","__GROUP__=results","__LICENSE__=results"	This entry holds the JSON tag value that is used during reconciliation for parsing multiple entries in a single payload.
nameAttributes	"__ACCOUNT__.firstName","__GROUP__.name","__LICENSE__.description"	This entry holds the name attribute for all the object classes that are handled by the connector. For example, the value `__ACCOUNT__.firstName` in decode implies that the `name` attribute of the connector for `__ACCOUNT__` object class is mapped to `firstName`, which is the corresponding name attribute for User account in the target system.

Table 3-2 (Cont.) Entries in the Lookup.GTM.Configuration Lookup Definition

Code Key	Decode	Description
opTypes	"__ACCOUNT__.licenseKeys.UPDATEOP=PUT","__ACCOUNT__.groupKey.UPDATEOP=PUT","__ACCOUNT__.CREATEOP=POST", "__ACCOUNT__.UPDATEOP=PUT","__ACCOUNT__.SEARCHOP=GET","__ACCOUNT__.DELETEOP=DELETE", "__ACCOUNT__.__LICENSE__.UPDATEOP=POST", "__ACCOUNT__.__LICENSE__.DELETEOP=DELETE"	This entry specifies the HTTP operation type for each object class supported by this 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).
relURIs	"__ACCOUNT__.licenseKeys.UPDATEOP=/admin/rest/v1/accounts/$(account_key)$/licenses/$(licenseKeys)$/users/$(__UID__)$", "__ACCOUNT__.CREATEOP=/admin/rest/v1/accounts/$(account_key)$/users", "__ACCOUNT__.UPDATEOP=/admin/rest/v1/accounts/$(account_key)$/users/$(__UID__)$", "__ACCOUNT__.SEARCHOP=/admin/rest/v1/accounts/$(account_key)$/users/$(Filter Suffix)$", "__ACCOUNT__.DELETEOP=/admin/rest/v1/accounts/$(account_key)$/users/$(__UID__)$", "__LICENSE__.SEARCHOP=/admin/rest/v1/accounts/$(account_key)$/licenses", "__GROUP__.SEARCHOP=/admin/rest/v1/accounts/$(account_key)$/groups", "__ACCOUNT__.__LICENSE__.ADDATTRIBUTE=/admin/rest/v1/accounts/$(account_key)$/licenses/$(__LICENSE__.key)$/users/$(__UID__)$", "__ACCOUNT__.__LICENSE__.REMOVEATTRIBUTE=/admin/rest/v1/accounts/$(account_key)$/licenses/$(__LICENSE__.key)$/users/$(__UID__)$", "__ACCOUNT__.groupKey.UPDATEOP=/admin/rest/v1/accounts/$(account_key)$/groups/$(groupKey)$/users/$(__UID__)$"	This entry holds the relative URL for all operations supported by the connector for each object class. For example, the `__ACCOUNT__.CREATEOP=/admin/rest/v1/accounts/$(account_key)$/users` value implies that `/admin/rest/v1/accounts/$(account_key)$/users` is the relative URL for all create provisioning operations that are performed on the `__ACCOUNT__` object class.
simpleMultivaluedAttribute	"__ACCOUNT__=primaryLicense"	This entry holds the name of the attributes that can hold multiple values. For example, the `primaryLicense` attribute holds the value as `License` because multiple licenses can be assigned to a user account.

Table 3-2 (Cont.) Entries in the Lookup.GTM.Configuration Lookup Definition

Code Key	Decode	Description
specialAttributeHandlling	"__ACCOUNT__.__LICENSE__.ADDATTRIBUTE=SINGLE","__ACCOUNT__.__LICENSE__.REMOVEATTRIBUTE=SINGLE", "__ACCOUNT__.__LICENSE__.CREATEOP=SINGLE","__ACCOUNT__.__LICENSE__.UPDATEOP=SINGLE", "__ACCOUNT__.licenseKeys.UPDATEOP=SINGLE", "__ACCOUNT__.groupKey.UPDATEOP=SINGLE"	This entry lists the **special attribute**, which is an attribute in an object class that can be managed only through a separate REST API endpoint rather than the same endpoint of the base object class. Values are sent to the target system in separate calls, one at a time. In addition, values are comma separated, and are in the following format: *OBJ_CLASS.ATTR_NAME*.*PROV_OP*=SINGLE For example, the `__ACCOUNT__.groupKey.UPDATEOP=SINGLE` value in decode implies that during an update provisioning operation, values for the `groupKey` attribute of the `__ACCOUNT__` object class is sent to the target system in separate calls, one at a time.
specialAttributeTargetFormat	"__ACCOUNT__.__GROUP__=managedGroupKeys","__ACCOUNT__.__LICENSE__=licenseKeys"	This entry lists the format in which a special attribute is present in the target system response. For example, the `__ACCOUNT__.__LICENSE__` attribute is present as `licenseKeys` in the target system response. Values are comma separated, and are presented in the following format: *OBJ_CLASS.ATTR_NAME*=*TARGET_FORMAT*
uidAttributes	"__ACCOUNT__.key","__GROUP__.key","__LICENSE__.key"	This entry holds the UID attribute for all the object classes that are handled by the connector. For example, the value `__ACCOUNT__.key` in decode implies that the `__UID__` attribute (that is, GUID) of the connector for `__ACCOUNT__` object class is mapped to `key`, which is the corresponding UID attribute for User account in the target system.

Table 3-2 (Cont.) Entries in the Lookup.GTM.Configuration Lookup Definition

Code Key	Decode	Description
User Configuration Lookup	Lookup.GTM.UM.Configuration	This entry holds the name of the lookup definition that stores configuration information used during user management operations.

3.1.5.1.2 Lookup.GTM.UM.Configuration

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

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

Table 3-3 Entries in the Lookup.GTM.UM.Configuration Lookup Definition

Code Key	Decode	Description
Provisioning Attribute Map	Lookup.GTM.UM.ProvAttrMap	This entry holds the name of the lookup definition that maps process form fields and target system attributes. This lookup definition is used during user provisioning operations.
Recon Attribute Map	Lookup.GTM.UM.ReconAttrMap	This entry holds the name of the lookup definition that maps resource object fields and target system attributes. This lookup definition is used during reconciliation.

3.1.5.1.3 Lookup.GTM.UM.ProvAttrMap

The Lookup.GTM.UM.ProvAttrMap lookup definition holds mappings between process form fields and target system attributes. This lookup definition is preconfigured, and is used during provisioning.

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

See Table 3-10 for a list of the default entries in this lookup definition.

3.1.5.1.4 Lookup.GTM.UM.ReconAttrMap

The Lookup.GTM.UM.ReconAttrMap lookup definition holds mappings between resource object fields and target system attributes. This lookup definition is preconfigured, and is used during reconciliation.

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

See Table 3-7 for a list of the default entries in this lookup definition.

3.1.5.1.5 Lookup.GTM.Locale

The Lookup.GTM.Locale lookup definition holds information about the supported locale codes for a target system account. This setting determines the display formats for date and time, users' names, addresses, and commas and periods in numbers.

This is a static lookup definition. This lookup definition is empty by default, and you must manually populate the entries of this lookup definition.

The following is the format in which you must add Code Key and Decode values in this lookup definition:

- **Code Key**: Standard ISO locale code for a target system account
- **Decode**: Name of the corresponding locale

Table 3-4 lists the sample entries in this lookup definition.

Table 3-4 Sample Entries in the Lookup.GTM.Locale Lookup Definition

Code	Decode
de_DE	Deutsch German
en_GB	International English
en_US	US English

3.1.5.2 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 license from the Licenses lookup field to specify the license being assigned to the user. **Lookup field synchronization** involves copying additions or changes made to specific fields in the target system to lookup definitions in Oracle Identity Manager.

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 populates these lookup fields with values from the corresponding lookup definitions.

The following lookup definitions are populated with values fetched from the target system by the scheduled jobs for lookup field synchronization:

- Lookup.GTM.Group
- Lookup.GTM.License

3.1.5.2.1 Lookup.GTM.Group

The Lookup.GTM.Group lookup definition holds values of all active groups that are available on the target system. The Group Name lookup field is populated with values from the Lookup.GTM.Group lookup definition, which is automatically created on Oracle Identity Manager after you deploy the connector.

You populate this lookup definition through lookup field synchronization performed using the GoToMeeting Group Lookup Reconciliation scheduled job.

The following is the format in which data is stored after lookup field synchronization:

- **Code Key:** *IT_RESOURCE_KEY~KEY*

  In this format:

  – *IT_RESOURCE_KEY* is the numeric code assigned to each IT resource in Oracle Identity Manager.

  – *KEY* is GUID of the group on the target system.

- **Decode:** *IT_RESOURCE_NAME~NAME*

  In this format:

  – *IT_RESOURCE_NAME* is the name assigned to the IT resource in Oracle Identity Manager.

  – *NAME* is the name of the group on the target system.

Table 3-5 lists sample entries in this lookup definition.

Table 3-5 Sample Entries in the Lookup.GTM.Group Lookup Definition

Code Key	Decode
87~123876986986986	GoToMeeting~Group1
87~123456786954321	GoToMeeting~Group2

3.1.5.2.2 Lookup.GTM.License

The Lookup.GTM.License lookup definition holds values of all account licenses that are available on the target system. The License lookup field is populated with values from the Lookup.GTM.License lookup definition, which is automatically created on Oracle Identity Manager after you deploy the connector.

You populate this lookup definition through lookup field synchronization performed using the GoToMeeting License Lookup Reconciliation scheduled job.

The following is the format in which data is stored after lookup field synchronization:

- **Code Key:** *IT_RESOURCE_KEY~KEY*

  In this format:

  – *IT_RESOURCE_KEY* is the numeric code assigned to each IT resource in Oracle Identity Manager.

  – *KEY* is the license key on the target system.

- **Decode:** *IT_RESOURCE_NAME~DESCRIPTION*

  In this format:

  – *IT_RESOURCE_NAME* is the name assigned to the IT resource in Oracle Identity Manager.

  – *DESCRIPTION* is the description of the license on the target system.

Table 3-6 lists sample entries in this lookup definition.

Table 3-6 Sample Entries in the Lookup.GTM.License Lookup Definition

Code Key	Decode
87~1234567891234567890	GoToMeeting~GoToMeeting Starter
87~5432167891234512345	GoToMeeting~GoToMeeting Pro

3.1.6 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 GoToMeeting Target Resource User Reconciliation scheduled job is used to initiate a reconciliation run. See Reconciliation Scheduled Jobs for the GoToMeeting Connector.

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

This section contains the following topics related to connector objects:

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

3.1.6.1 User Fields for Target Resource Reconciliation

The Lookup.GTM.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 3-7 lists the entries in this lookup definition.

Table 3-7 Entries in the Lookup.GTM.UM.ReconAttrMap Lookup Definition

Code Key	Decode
Email	email
First Name	__NAME__
Group[LOOKUP]	groupKey
Last Name	lastName

Table 3-7 (Cont.) Entries in the Lookup.GTM.UM.ReconAttrMap Lookup Definition

Code Key	Decode
Locale	locale
PrimaryLicense[LOOKUP]	PARENT.licenseKeys
Licenses~License Name[LOOKUP]	CHILD.licenseKeys
key	__UID__

3.1.6.2 Reconciliation Rules for Target Resource Reconciliation

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: GoToMeeting User Recon Rule

Rule element: Email Equals Email

In this rule element:

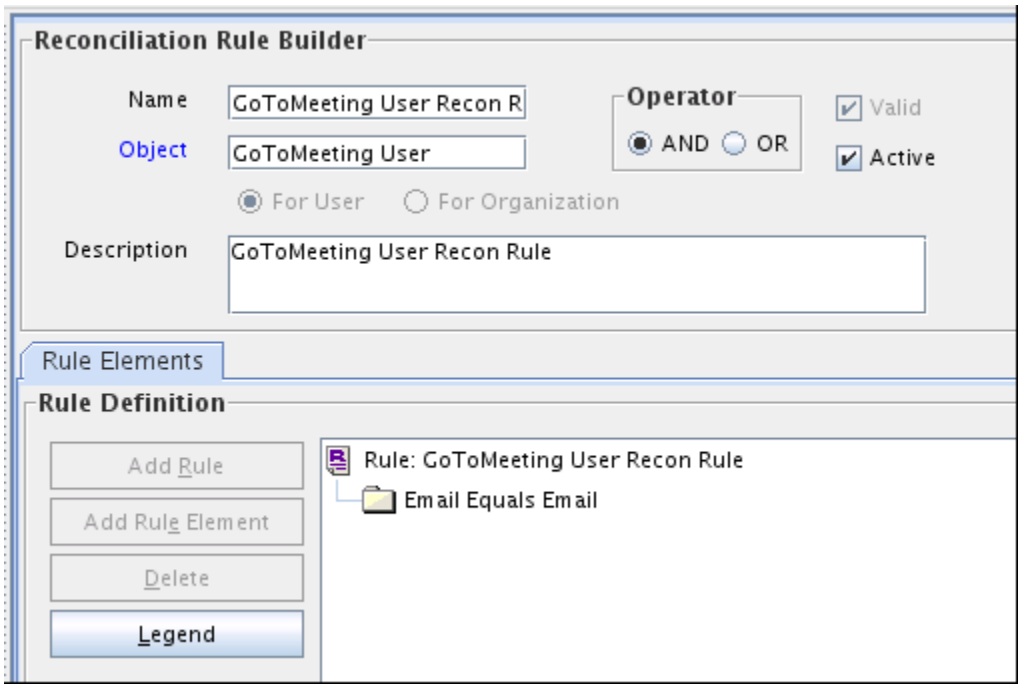
- The first Email reference is the email address attribute of the OIM user.
- The second Email reference is the email address attribute of the user in GoToMeeting.

3.1.6.3 Viewing Reconciliation Rules for Target Resource Reconciliation

You can view reconciliation rules by using Oracle Identity Manager Design Console. To view reconciliation rules for target resource reconciliation:

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

Figure 3-2 GoToMeeting User Recon Rule



3.1.6.4 Reconciliation Action Rules for Target Resource Reconciliation

Reconciliation action rules define the actions the connector must perform based on the reconciliation rules defined for Users. Table 3-8 lists the rule condition and the corresponding action to be performed during target resource reconciliation.

Table 3-8 Action Rules for Target Resource Reconciliation

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

3.1.6.5 Viewing Reconciliation Action Rules for Target Resource Reconciliation

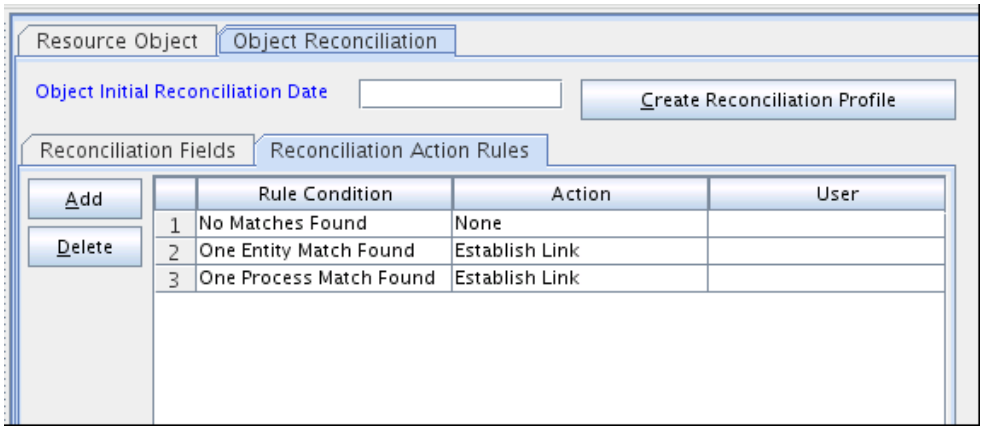
You can view reconciliation action rules on the Object Reconciliation tab of a resource object in Oracle Identity Manager Design Console.

To view reconciliation action rules for target resource reconciliation:

1. Log in to Oracle Identity Manager Design Console.
2. Expand **Resource Management**.
3. Double-click **Resource Objects**.
4. Search for and open the **GoToMeeting 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 3-3 Reconciliation Action Rules for Target Resource Reconciliation



3.1.7 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.

This section contains the following topics:

- Provisioning Functions
- User Fields for Provisioning

3.1.7.1 Provisioning Functions

These are the supported provisioning functions and the adapters that perform these functions for the connector. The Adapter column in Table 3-9 gives 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 3-9 User Provisioning Functions

Function	Adapter
Create User	adpGTMCREATEOBJECT
Update User	adpGTMUPDATEOBJECT
Delete user	adpGTMDELETEOBJECT
Update child table values	adpGTMUPDATECHILDDATA

Table 3-9 (Cont.) User Provisioning Functions

Function	Adapter
Add child table values	adpGTMADDCHILDOBJECT
Remove child table values for a user	adpGTMREMOVECHILDOBJECT

3.1.7.2 User Fields for Provisioning

The Lookup.GTM.UM.ProvAttrMap lookup definition holds the user fields for provisioning. This lookup definition holds mapping between process form fields and target system attributes.

Table 3-10 Entries in the Lookup.GTM.UM.ProvAttrMap Lookup Definitions

Code Key	Decode
Email	email
First Name	__NAME__
Group[LOOKUP]	groupKey
Last Name	lastName
Locale	locale
PrimaryLicense[LOOKUP]	licenseKeys
UD_GTM_LIC~License Name[LOOKUP]	__LICENSE__~__LICENSE__~key
key	__UID__

3.1.8 Roadmap for Deploying and Using the GoToMeeting 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 GoToMeeting Connector describes procedures that you must perform on Oracle Identity Manager and the target system during each stage of connector deployment.
- Using the GoToMeeting Connector describes guidelines on using the connector, and explains procedures to configure reconciliation runs and perform provisioning operations.
- Extending the Functionality of the GoToMeeting Connector describes procedures that you can perform if you want to extend the functionality of the connector.
- Known Issues and Workarounds for the GoToMeeting Connector lists known issues associated with this release of the connector.
- Files and Directories on the GoToMeeting Connector Installation Media lists the files and directories that comprise the connector installation media.

3.2 Deploying the GoToMeeting Connector

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

The following topics provide details on these stages:

- Preinstallation
- Installation
- Postinstallation

3.2.1 Preinstallation

Preinstallation involves creating a developer application and obtaining the consumer key and consumer secret values. It also involves generating the access token and refresh token values and obtaining the account key for your developer account.

To obtain these values, perform the following tasks on the target system:

1. Set up the Developer Sandbox, create a developer application (GoToMeeting App), and obtain the consumer key and consumer secret values that are generated after the application is created.

   You provide the consumer key and consumer secret values for the clientId and clientSecret parameters respectively while configuring the IT resource.

2. Generate the access token and refresh token values using the consumer key value that you obtained in Step 1.

   You provide the access token and refresh token values for the customAuthHeaders parameter while configuring the IT resource. You manually generate these tokens for the first time. Subsequently, the GoToMeeting Update Access Token scheduled job is run to renew these values in a periodic manner.

   In addition to the access token and refresh token values, the account key value is displayed for your Developer Sandbox account. You provide the account key value for the uriPlaceHolder parameter while configuring the IT resource.

The detailed instructions for performing these preinstallation tasks are available in the GoToMeeting Developer Center documentation at https://goto-developer.logmeininc.com/.

3.2.2 Installation

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

The following topics provide details on installing the connector:

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

3.2.2.1 Understanding Installation of the GoToMeeting 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. For information about installing, configuring, and running the Connector Server, and then installing the connector using a Connector Server, see Using an Identity Connector Server in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager*.

3.2.2.2 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.

To run the Connector Installer:

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

   OIM_HOME`/server/ConnectorDefaultDirectory`

   > **Note:**
   >
   > If this is the first time you are running the Connector Installer for deploying the connector bundle in a Connector Server, then place the bundle in the connector server bundle directory.

2. Log in to Oracle Identity System Administration.
3. In the left pane, under System Management, click **Manage Connector**.
4. In the Manage Connector page, click **Install**.
5. From the Connector List list, select **GoToMeeting Connector *RELEASE_NUMBER***. This list displays the names and release numbers of connectors whose installation files you copy into the default connector installation directory: *OIM_HOME*`/server/ConnectorDefaultDirectory`

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

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

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

c. From the Connector List list, select **GoToMeeting 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 Connector XML Files (Using Deployment Manager)
   c. Compilation of Adapter Definitions

   On successful completion of a task, a check mark appears for the task. If a task fails, then an X mark and a message stating the reason for failure are displayed. If a task fails, then make the required correction and perform one of the following steps:
   a. To retry the installation, click **Retry**.
   b. To cancel the installation and restart the installation process, click **Cancel** and begin the procedure from Step 3.
8. Click **Exit** to finish the installation procedure.

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

   In addition, a list of the steps that you must perform after the installation appears. These steps are as follows:
   a. Configuring the IT resource for the connector.

      The procedure to configure the IT resource is described later in this guide.
   b. Configuring the scheduled tasks that are created when you installed the connector.

      The procedure to configure these scheduled tasks is described later in this guide.

   When you run the Connector Installer, it copies the connector files and external code files to destination directories on the Oracle Identity Manager host computer. These files are listed in Files and Directories on the GoToMeeting Connector Installation Media.

3.2.2.3 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 enable the connector to connect Oracle Identity Manager with your target system.

This section contains the following topics:

- IT Resource Parameters
- Specifying Values for IT Resource Parameters

3.2.2.3.1 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 3-11 displays each parameter of the GoToMeeting IT resource in an alphabetical order.

Table 3-11 IT Resource Parameters

Parameter	Description
Configuration Lookup	Name of the lookup definition that stores configuration information used during reconciliation and provisioning. Default value: `Lookup.GTM.Configuration` Do *not* modify the value of the parameter.
Connector Server Name	If you have deployed the GoToMeeting connector in the Connector Server, then enter the name of the IT resource for the Connector Server.
authenticationType	Type of authentication that is used by your target system. This connector supports authenticating to the target system by using OAuth 2.0 custom authentication type. Default value: `Other` Do *not* modify the value of the parameter.
clientId	Enter the consumer key value that is generated after creating the developer application. Sample Value: `ABCDEbkTacBC7emdnbABCDEFa96DsEYN` See Preinstallation for more information on obtaining the consumer key value.
clientSecret	Enter the consumer secret value that is generated after creating the developer application. Sample value: `AB9CDoOOabCo2l03` See Preinstallation for more information on obtaining the consumer secret value.

customAuthHeaders	Enter the access token and refresh token values in the following format:
	`"access_token=ACCESSTOKEN", "refresh_token=REFRESHTOKEN"`
	In this format, replace *ACCESSTOKEN* with the access token value and *REFRESHTOKEN* with the refresh token value. These values are generated after setting up the Developer Sandbox.
	Sample value:
	`access_token=ABabEXAMPLe0Q0ZjABCabc0AbAbC","refresh_token=hABCfd9oABc6abcDeFGabcdXhwRMIHav"`
	See Preinstallation for more information on obtaining the access token and refresh token values.

Table 3-11 (Cont.) IT Resource Parameters

Parameter	Description
host	Enter the host name of your target system. Sample value: `api.getgo.com`
port	Enter the port number at which the target system is listening.
proxyHost	Enter the name of the proxy host that is used to connect to an external target. Sample value: `www.example.com`
proxyPort	Enter the proxy port number.
proxyUser	Enter the proxy user name of the target system user account that Oracle Identity Manager uses to connect to the target system.
proxyPassword	Enter the password of the proxy user ID of the target system user account that Oracle Identity Manager uses to connect to the target system.
sslEnabled	If the target system requires SSL connectivity, then set the value of this parameter to `true`. Otherwise set the value to `false`. Default value: `true`
uriPlaceHolder	Enter the account key value that is displayed while generating the access token and refresh token values for your Developer Sandbox account. Sample value: `"account_key;5253092000266355206"` See Preinstallation for more information on obtaining the account key value.

3.2.2.3.2 Specifying Values for 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 GoToMeeting IT resource is automatically created when you run the Connector Installer, and you must specify values for the parameters of the IT resource.

To specify values for the parameters of the IT resource:

1. Log in to Identity System Administration.
2. 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*.
3. In the left pane, under Configuration, click **IT Resource**.
4. In the **IT Resource Name** field on the Manage IT Resource page, enter `GoToMeeting` and then click **Search**.
5. Click the Edit icon for the IT resource.
6. From the list at the top of the page, select **Details and Parameters**.

7. Specify values for the parameters of the IT resource. Table 3-11 describes each parameter.
8. To save the values, click **Update**.

3.2.3 Postinstallation

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

The postinstallation tasks are divided across the following sections:

- 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 GoToMeeting Connector
- Configuring SSL for the GoToMeeting Connector

3.2.3.1 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 the 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

3.2.3.1.1 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*.

3.2.3.1.2 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 connector that you want to associate the form with. In addition, select the **Generate Entitlement Forms** check box.

3.2.3.1.3 Associating the Form with the Application Instance

By default, an application instance named **GoToMeeting 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. See Modifying Application Instances in *Oracle Fusion Middleware Administering Oracle Identity Manager*.

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.

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. On the Catalog page, check for the Application Instance UI (form fields) and ensure that it appears correctly.
5. Publish the application instance only if everything appears correctly. Otherwise, fix the issues and then publish the application instance.

   See Managing Organizations Associated With Application Instances in *Oracle Fusion Middleware Administering Oracle Identity Manager* for instructions on publishing an application instance to an organization.

3.2.3.1.4 Publishing a Sandbox

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

1. In Identity System Administration, deactivate the sandbox.
2. Log out of Identity System Administration.
3. Log in to Identity Self Service using the xelsysadm user credentials and then activate the sandbox that you deactivated in Step 1.
4. In the Catalog, ensure that the GoToMeeting 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*.

3.2.3.1.5 Harvesting Entitlements and Sync Catalog

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

To harvest entitlements and sync catalog:

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

> **See Also:**
>
> Predefined Scheduled Tasks in *Oracle Fusion Middleware Administering Oracle Identity Manager* for a description of the Entitlement List and Catalog Synchronization Job scheduled jobs

3.2.3.1.6 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*.

3.2.3.2 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. The resource bundles are available in the connector installation media.

To localize a field label that is added to 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 GoToMeeting 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_GTM_USER_FIRSTNAME__c_description']}">
<source>First Name</source>
<target/>
</trans-unit>
<trans-unit
id="sessiondef.oracle.iam.ui.runtime.form.model.GoToMeetingForm.e
ntity.GoToMeetingFormEO.UD_GTM_USER_ FIRSTNAME__c_LABEL">
<source>First Name</source>
<target/>
</trans-unit>
```

In this text, *GoToMeetingForm* is the current form instance name associated with the GoToMeeting application instance.

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

```
global.udf.UD_GTM_USER_FIRSTNAME=\u540D
```

e. Replace the original code shown in Step 6 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
.userEO.UD_GTM_USER_FIRSTNAME__c_description']}">
<source>First Name</source>
<target>\u540D</target>
</trans-unit>
<trans-unit
id="sessiondef.oracle.iam.ui.runtime.form.model.GoToMeetingForm.e
ntity.GoToMeetingFormEO.UD_GTM_USER_ FIRSTNAME__c_LABEL">
<source>First Name</source>
<target>\u540D</target>
</trans-unit>
```

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

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

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

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

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

3.2.3.3 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:

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

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

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.

3.2.3.4 Managing Logging for the GoToMeeting Connector

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

The following topics provide detailed information about logging:

- Understanding Log Levels
- Enabling Logging

3.2.3.4.1 Understanding Log Levels

When you enable logging, Oracle Identity Manager automatically stores in a log file information about events that occur during the course of provisioning and reconciliation operations. ODL is the principle logging service used by Oracle Identity Manager and is based on java.util.Logger.

To specify the type of event for which you want logging to take place, you can set the log level to one of the following:

- SEVERE.intValue()+100

  This level enables logging of information about fatal errors.

- SEVERE

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

- WARNING

  This level enables logging of information about potentially harmful situations.

- INFO

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

- CONFIG

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

- FINE, FINER, FINEST

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

These message types are mapped to ODL message type and level combinations as shown in Table 3-12.

Table 3-12 Log Levels and ODL Message Type:Level Combinations

Java Level	ODL Message Type:Level
SEVERE.intValue()+100	INCIDENT_ERROR:1
SEVERE	ERROR:1
WARNING	WARNING:1
INFO	NOTIFICATION:1
CONFIG	NOTIFICATION:16
FINE	TRACE:1
FINER	TRACE16
FINEST	TRACE32

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.

3.2.3.4.2 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='GoToMeeting-handler'
level='[LOG_LEVEL]'class='oracle.core.ojdl.logging.ODLHandlerFact
ory'>
<property name='logreader:' value='off'/>
<property name='path' value='[FILE_NAME]'/>
<property name='format' value='ODL-Text'/>
```

```
<property name='useThreadName' value='true'/>
<property name='locale' value='en'/> <property
name='maxFileSize' value='5242880'/>
<property name='maxLogSize' value='52428800'/>
<property name='encoding' value='UTF-8'/>
</log_handler>


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

b. Replace both occurrences of [LOG_LEVEL] with the ODL message type and level combination that you require. Table 3-12 lists the supported message type and level combinations. Similarly, replace [FILE_NAME] with the full path and name of the log file in which you want log messages to be recorded. The following blocks show sample values for **[LOG_LEVEL]** and **[FILE_NAME]:**

```
<log_handler name='GoToMeeting-handler' level='NOTIFICATION:1'
class='oracle.core.ojdl.logging.ODLHandlerFactory'>
<property name='logreader:' value='off'/>
<property name='path'
value='F:\MyMachine\middleware\user_projects\domains\base_domain1
\servers\o im_server1\logs\oim_server1-diagnostic-1.log'/>
<property name='format' value='ODL-Text'/>
<property name='useThreadName' value='true'/>
<property name='locale' value='en'/>
<property name='maxFileSize' value='5242880'/>
<property name='maxLogSize' value='52428800'/>
<property name='encoding' value='UTF-8'/>
</log_handler>

<logger name="ORG.IDENTITYCONNECTORS.GoToMeeting"
level="NOTIFICATION:1"useParentHandlers="false">
<handler name="GoToMeeting-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.

3.2.3.5 Configuring SSL for the GoToMeeting Connector

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

To configure SSL:

1. Obtain the SSL certificate by obtaining the public key certificate of the target system.
2. Copy the public key certificate of the target system 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 -keystore WEBLOGIC_HOME/server/lib/DemoTrust.jks -file
   CERT_FILE_NAME -storepass PASSWORD
   ```

   In this command:

   - *CERT_FILE_NAME* is the full path and name of the certificate file
   - *PASSWORD* is the password of the keystore.

   The following is a sample value for this command:

   ```
   keytool -import -keystore WEBLOGIC_HOME/server/lib/DemoTrust.jks -
   file /home/target.cert -storepass DemoTrustKeyStorePassPhrase
   ```

> **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.

3.3 Using the GoToMeeting Connector

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

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

- Configuring Reconciliation for the GoToMeeting Connector
- Configuring Scheduled Jobs

- Performing Provisioning Operations
- Uninstalling the GoToMeeting Connector

3.3.1 Scheduled Jobs for Lookup Field Synchronization

The GoToMeeting Group Lookup Reconciliation and GoToMeeting License Lookup Reconciliation scheduled jobs are used for lookup field synchronization. Values fetched by these scheduled jobs from the target system are populated in the Lookup.GTM.Group and Lookup.GTM.License lookup definitions, respectively.

Table 3-13 Attributes of the Scheduled Jobs for Lookup Field Synchronization

Attribute	Description
Code Key Attribute	Name of the connector attribute that is used to populate the Code Key column of the lookup definition (specified as the value of the Lookup Name attribute). Default value: `__UID__`
Decode Attribute	Name of the connector attribute that is used to populate the Decode column of the lookup definition (specified as the value of the Lookup Name attribute). Default value: `__NAME__`
IT Resource Name	Name of the IT resource for the target system installation from which you want reconcile user records. Default value: `GoToMeeting`
Lookup Name	Enter the 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 that you are using, the default values are as follows: • For GoToMeeting Group Lookup Reconciliation: `Lookup.GTM.Group` • For GoToMeeting License Lookup Reconciliation: `Lookup.GTM.License` If you create a copy of any of these lookup definitions, then enter the name of that new lookup definition as the value of the Lookup Name attribute.
Object Type	Enter the type of object you want to reconcile. Depending on the scheduled job that you are using, the default values are as follows: • For GoToMeeting Group Lookup Reconciliation: `__GROUP__` • For GoToMeeting License Lookup Reconciliation: `__LICENSE__`

3.3.2 Configuring Reconciliation for the GoToMeeting Connector

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

This section provides details on the following topics related to configuring reconciliation:

- Full Reconciliation
- Performing Limited Reconciliation
- Reconciling Large Number of Records
- Reconciliation Scheduled Jobs for the GoToMeeting Connector

3.3.2.1 Performing Full Reconciliation

Full reconciliation involves reconciling all existing user records from the target system into Oracle Identity Manager. After you deploy the connector, you must first perform full reconciliation.

To perform a full reconciliation run, ensure that no value is specified for the Filter attribute of the scheduled job for reconciling users. 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.

3.3.2.2 Performing Limited Reconciliation

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

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.

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

> **Note:**
>
> 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 limited reconciliation. See Reconciling Large Number of Records.

For more information on GoToMeeting filters, see the filtering information related to GoToMeeting Administration REST API Implementation on the GoToMeeting Developer Center page at https://goto-developer.logmeininc.com/.

3.3.2.3 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, you can fetch all the records into Oracle Identity Manager using the Flat File connector. The Flat File connector consumes information in a flat file, and generates connector metadata using the metadata generation utility.

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. See Creating a Schema File in *Oracle Identity Manager Connector Guide for Flat File*.
4. Install the Flat File connector. See Running the Connector Installer in *Oracle Identity Manager Connector Guide for Flat File*.
5. Configure the Flat File IT resource. See Configuring the IT Resource in *Oracle Identity Manager Connector Guide for Flat File*.
6. 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 `GoToMeeting` and **Target Resource Object Name** to `GoToMeeting User`.

   See Flat File Accounts Loader and IT_RES_NAME Flat File Accounts Loader in *Oracle Identity Manager Connector Guide for Flat File* for information about the attributes of the Flat File Accounts Loader scheduled job.

3.3.2.4 Reconciliation Scheduled Jobs for the GoToMeeting Connector

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.

This section discusses the following scheduled jobs that you can configure for reconciliation:

- GoToMeeting User Reconciliation
- GoToMeeting Update Access Token

3.3.2.4.1 GoToMeeting User Reconciliation

The GoToMeeting User Reconciliation scheduled job is used to reconcile user account data from the target system in the target resource (account management) mode of the connector.

Table 3-14 Attributes of the GoToMeeting User Reconciliation Scheduled Job

Attribute	Description
Filter Suffix	Enter the search filter for fetching user records from the target system during a reconciliation run. See Performing Limited Reconciliation. Sample value: `"?filter=firstName%20%3D%20%22.*UserA.*%22"`
IT Resource Name	Enter the name of the IT resource for the target system installation from which you want to reconcile user records. Default value: `GoToMeeting`
Object Type	This attribute holds the name of the object type for the reconciliation run. Default value: `User` Do *not* change the default value.
Resource Object Name	Name of the resource object against which reconciliation runs are performed. Default value: `GoToMeeting User` Do *not* change the default value.

3.3.2.4.2 GoToMeeting Update Access Token

The GoToMeeting Update Access Token scheduled job is used to automatically refresh the access token value (configured as part of the IT resource) before it expires.

Table 3-15 Attributes of the GoToMeeting Update Access Token Scheduled Job

Attribute	Description
Access Token Endpoint	This attribute holds the GoToMeeting Administration API endpoint to get a new access token value. Default value: `https://api.getgo.com/oauth/access_token` Do *not* modify the value of this attribute.
IT Resource Name	Enter the name of the IT resource for the target system installation from which you want to reconcile user records. Default value: `GoToMeeting`
Task Name	This attribute holds the name of the scheduled task. Default value: `GoToMeeting Update Access Token` You must *not* change the default value.

3.3.3 Configuring Scheduled Jobs

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

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

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

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

> **Note:**
>
> - Values (either default or user-defined) must be assigned to all the attributes. If even a single attribute value is left empty, then reconciliation is not performed.
> - See Reconciliation Scheduled Jobs for the GoToMeeting Connector for the list of scheduled tasks and their attributes.

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

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

3.3.4 Performing Provisioning Operations

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

To perform provisioning operations in Oracle Identity Manager:

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

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

3.3.5 Uninstalling the GoToMeeting Connector

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

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

3.4 Extending the Functionality of the GoToMeeting Connector

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

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

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

3.4.1 Adding 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 or group attributes for reconciliation.

By default, the attributes listed in Table 3-7 are mapped for reconciliation between Oracle Identity Manager and the target system.

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 Definitions for Provisioning
- Performing Changes in a New UI Form

3.4.1.1 Adding New Attributes on the Process Form

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

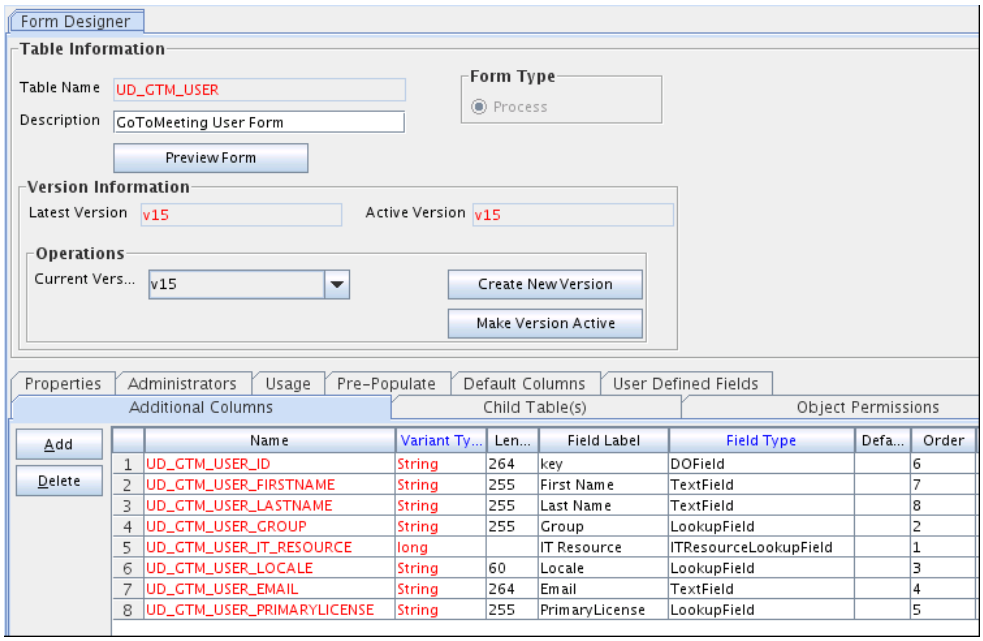
To add a new attribute on the process form:

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_GTM_USER** process form for users.
4. Click **Create New Version**, and then **Add**.
5. Enter the details of the field.

For example, if you are adding the **Last Name** field, enter `UD_GTM_USER_LASTNAME` 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 figure shows the new field added to the process form.

Figure 3-4 Adding a New Field on the Process Form



3.4.1.2 Adding Attributes to the Resource Object

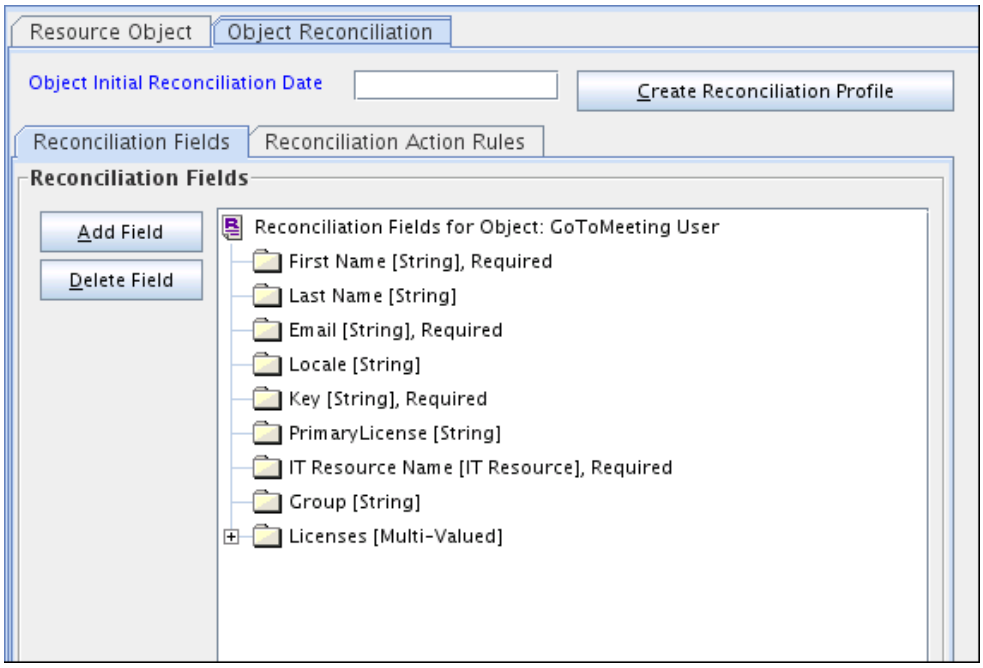
You can add the new attribute to the resource object in the Resource Objects section of Oracle Identity Manager Design Console.

To add the new attribute to the list of reconciliation fields in the resource object:

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

   For example, enter `Last Name` in the 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 figure shows the new reconciliation field added to the resource object:

Figure 3-5 Newly Added Reconciliation Field



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

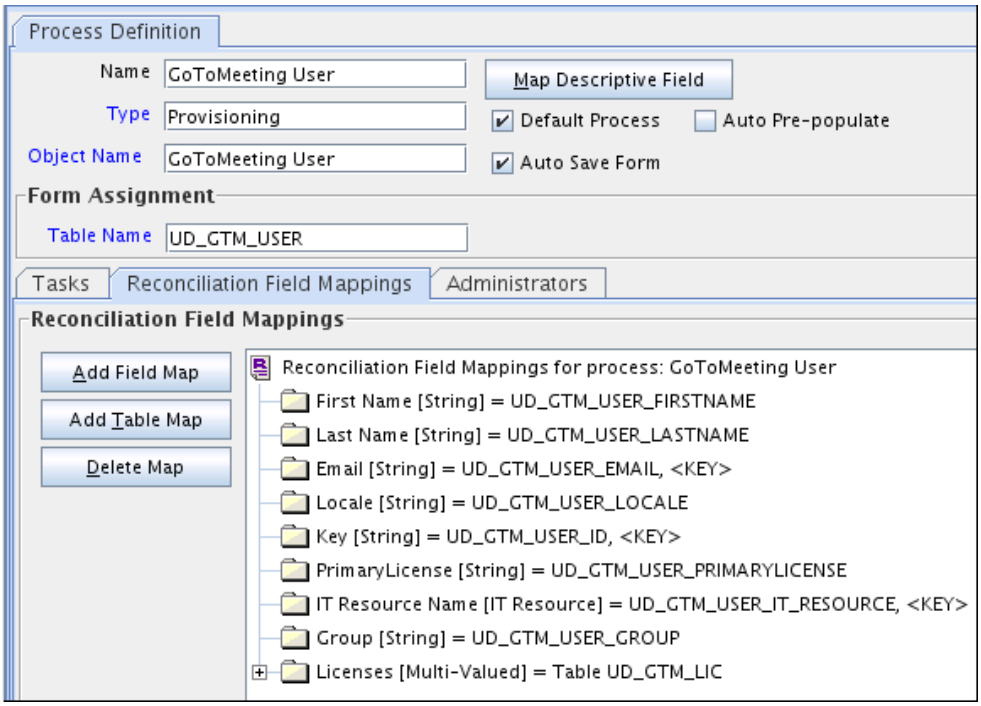
3.4.1.3 Creating Reconciliation Field Mapping

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

To create reconciliation field mapping for the new attribute in the process definition:

1. Expand **Process Management** and double-click **Process Definition**.
2. Search for and open the **GoToMeeting 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 select the column for the attribute. For example, select **UD_GTM_USER_LASTNAME**.
6. Click the Save icon. The following figure shows the new reconciliation field mapped to a process data field in the process definition:

Figure 3-6 New Reconciliation Field Mapped to a Process Data Field in the Process Definition



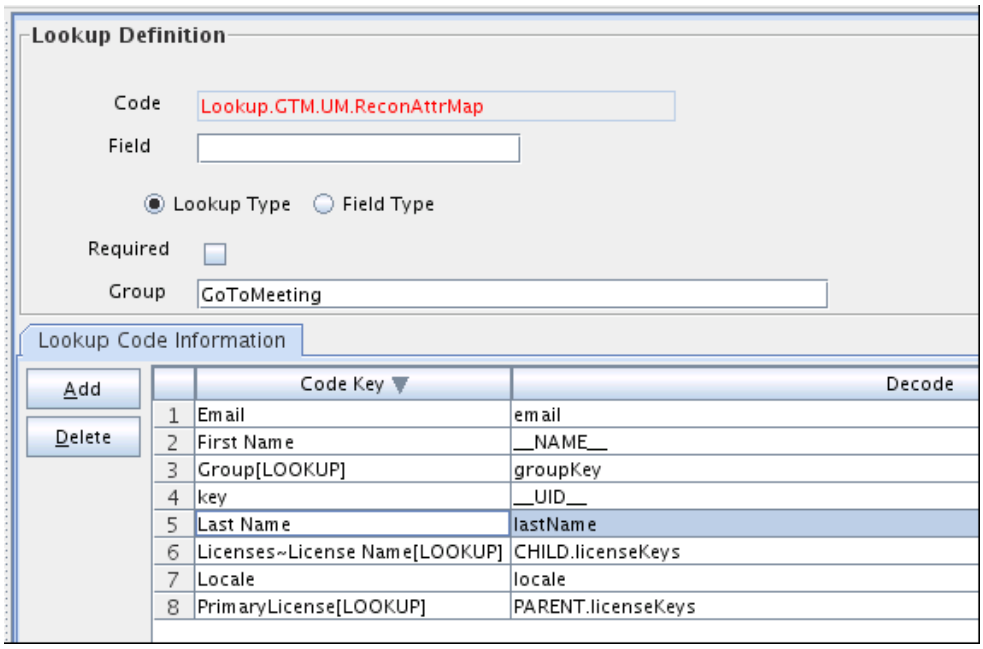
3.4.1.4 Creating Entries in Lookup Definitions for Reconciliation

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

To create an entry for the newly added attribute in the lookup definition:

1. Expand **Administration**.
2. Double-click **Lookup Definition**.
3. Search for and open the **Lookup.GTM.UM.ReconAttrMap** lookup definition for users.
4. Click **Add** and enter the Code Key and Decode values for the field. The Code Key value must be the name of the field in the resource object.
5. Click the Save icon. The following figure shows the entry added to the lookup definition:

Figure 3-7 Newly Added Entry to Lookup Definition



3.4.1.5 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.

To perform changes in a new UI form:

1. Log in to Identity System Administration.
2. 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*.
3. Create a new UI form to view the newly added field along with the rest of the fields. See Creating Forms By Using the Form Designer in *Oracle Fusion Middleware Administering Oracle Identity Manager*.
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 in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager*.

3.4.2 Adding 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 3-10.

The following topics provide details on adding new user attributes for provisioning:

- Adding New Attributes for Provisioning

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

3.4.2.1 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.

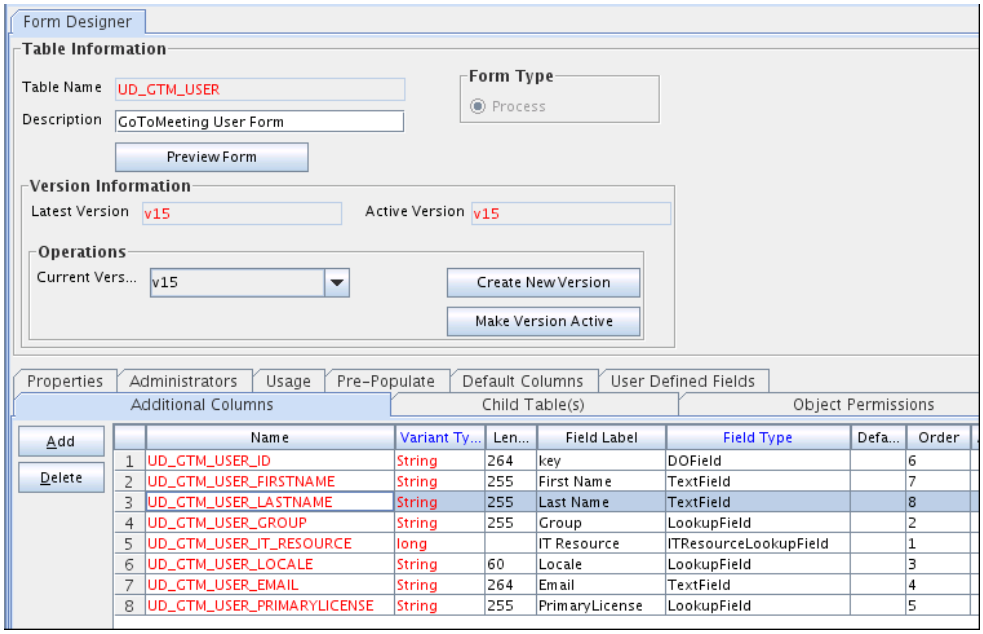
> **Note:**
>
> If you have already added an attribute for reconciliation, then you need not repeat steps performed as part of that procedure.

To add a new attribute on the process form:

1. Log in to the Oracle Identity Manager Design Console.
2. Expand **Development Tools** and double-click **Form Designer**.
3. Search for and open one the **UD_GTM_USER** 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 `Last Name` field, enter `UD_GTM_USER_LASTNAME` 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**.

Figure 3-8 Newly Added Field



3.4.2.2 Creating Entries in Lookup Definitions for Provisioning

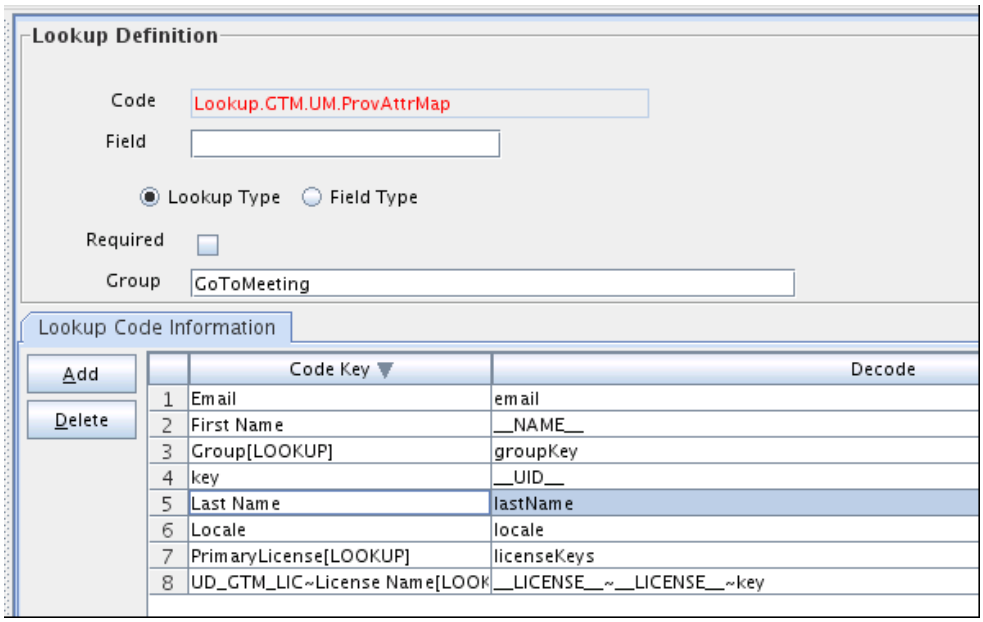
You can create an entry for the newly added attribute in the lookup definition that holds attribute mappings for provisioning.

To create an entry for the newly added attribute in the lookup definition:

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

   For example, enter `Last Name` in the Code Key column and then enter `lastName` in the Decode column. The following figure shows the entry added to the lookup definition:

Figure 3-9 Newly Added Entry to the Lookup Definition



3.4.2.3 Creating a Task to Enable Update Operations

You create a task to enable updates on the new user or group attribute during provisioning operations. If you do not perform this procedure, you cannot modify the value of the attribute after you set a value for it during the Create User provisioning operation.

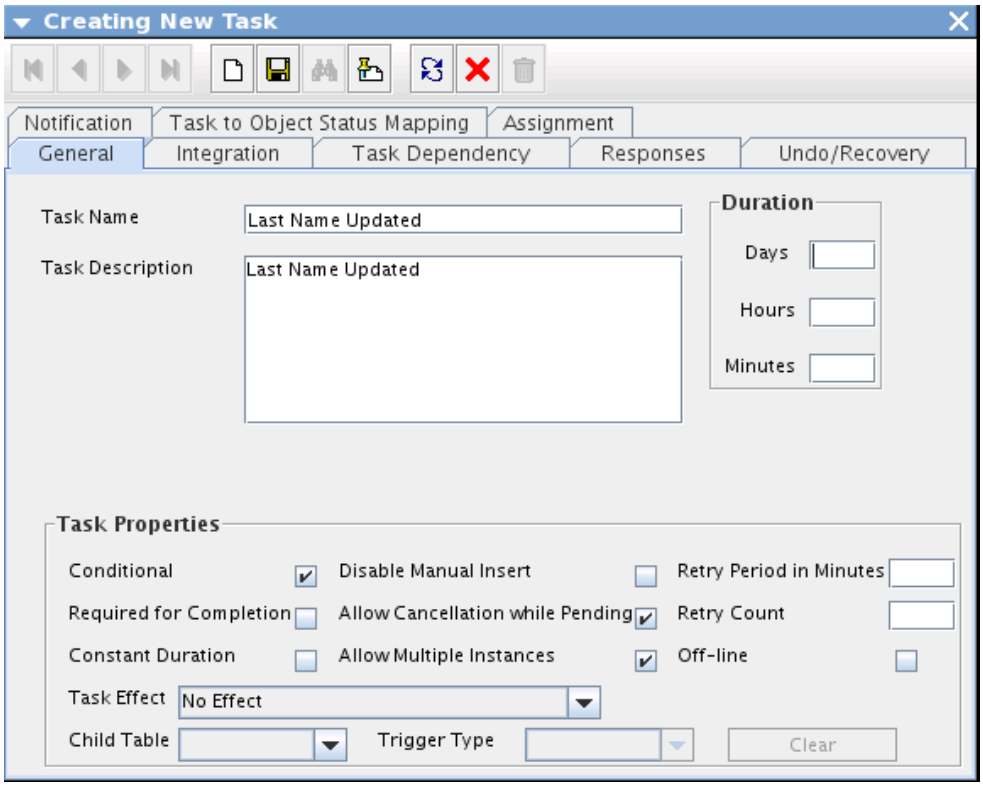
To enable the update of the attribute during provisioning operations, add a process task for updating the new user or group attribute as follows:

1. Expand **Process Management** and double-click **Process Definition**.
2. Search for and open the **GoToMeeting 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 figure shows the new task added to the process definition:

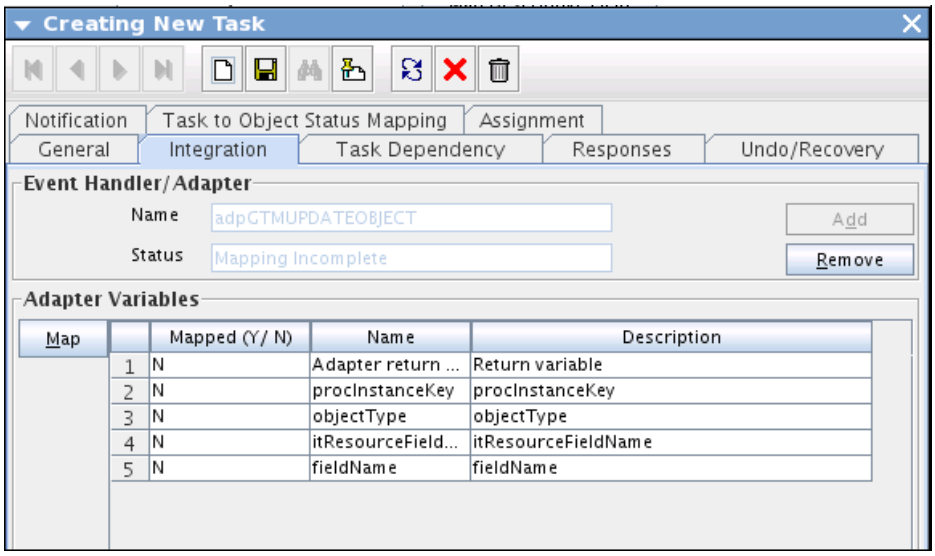
Figure 3-10 Newly Added Task to the Process Definition



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 **adpGTMUPDATEOBJECT**.
   d. Click **Save**, and close the dialog box.

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

Figure 3-11 List of Adapter Variables



7. In the Adapter Variables region, click the **procInstanceKey** variable.
8. In the dialog box that is displayed, map the adapter variable as follows:
   a. Click **Map**. The Data Mapping for Variable window is displayed.
   b. Complete the following fields:
      - **Variable Name**: procInstanceKey
      - **Map To**: Process Data
      - **Qualifier**: Process Instance
9. Click Save and close the dialog box.

   The mapping status for the adapter variable changes from N to Y. This indicates that the adapter variable has been mapped.
10. If you are enabling update provisioning operations for a User attribute, then repeat Step 7 through Step 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_GTM_USER_IT_RESOURCE
attributeFieldName	Literal	String	Last Name

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.

3.4.2.4 Replicating Form Designer Changes to a New UI Form

To replicate all changes made to the Form Designer of the Design Console in a new UI form:

1. Log in to Identity System Administration.
2. 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*.
3. Create a new UI form to view the newly added field along with the rest of the fields. See Creating Forms By Using the Form Designer in *Oracle Fusion Middleware Administering Oracle Identity Manager*.
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 in *Oracle Fusion Middleware Developing and Customizing Applications for Oracle Identity Manager*.

3.4.3 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 First Name attribute to ensure that it does not contain the number sign (#). In addition, you can validate data entered in the First Name field on the process form so that the number sign (#) is not sent to the target system during provisioning operations. For data that fails the validation check, the following message is displayed or recorded in the log file: Validation failed for attribute *ATTRIBUTE_NAME*.

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 First 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.
        */
        String sFirstName=(String) hmUserDetails.get(field);
        if( sFirstName.contains("#")){
                return false;
        }
        return true;
}
```

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.GTM.UM.ReconValidation**.
   c. In the Code Key column, enter the resource object field name that you want to validate. For example, `First Name`.
      In the Decode column, enter the class name. For example, `org.identityconnectors.GTM.extension.GTMValidator`.
   d. Save the changes to the lookup definition.
   e. Search for and open the Lookup.GTM.UM.Configuration lookup definition.
   f. In the Code Key column, enter `Recon Validation Lookup`. In the Decode column, enter `Lookup.GTM.UM.ReconValidation`.

g. Save the changes to the lookup definition.

5. If you have 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 by the name Lookup.GTM.UM.ProvValidation.
   c. In the Code Key column, enter the process form field name. In the Decode column, enter the class name.
   d. Save the changes to the lookup definition.
   e. Search for and open the Lookup.GTM.UM.Configuration lookup definition.
   f. In the Code Key column, enter `Provisioning Validation Lookup`. In the Decode column, enter `Lookup.GTM.UM.ProvValidation.`
   g. Save the changes to the lookup definition.
   h. Purge the cache to ensure that the changes are reflected in Oracle Identity Manager.

3.4.4 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 First 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 First 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 sFirstName= (String)hmUserDetails.get("First
Name");
                String sLastName= (String)hmUserDetails.get("Last
Name");
                String sFullName=sFirstName+"."+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. If you created the Java class for transforming a process form field for reconciliation, then:
   a. Log in to the Design Console.
   b. Create a lookup definition named **Lookup.GTM.UM.ReconTransformation**.
   c. In the Code Key column, enter the resource object field name on which you want to apply transformation. For example, First Name. In the Decode column, enter the name of the class that implements the transformation logic. For example, oracle.iam.connectors.common.transform.TransformAttribute.
   d. Save the changes to the lookup definition.
5. Add an entry in the **Lookup.GTM.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.GTM.UM.Configuration** lookup definition.

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

d. Save the changes to the lookup definition.

e. Purge the cache to ensure that the changes are reflected in Oracle Identity Manager.

3.4.5 Configuring the GoToMeeting Connector for Multiple Installations of the Target System

You must create copies of the connector to configure it for multiple installations of the target system.

The following example illustrates this requirement:

The London and New York offices of Example Multinational Inc. have their own installations of the target system. 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*.

3.4.6 Defining the GoToMeeting Connector

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

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

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

The following events take place when you define a connector:

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

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

3.5 Known Issues and Workarounds for the GoToMeeting Connector

These are the known issues and workarounds associated with this release of the GoToMeeting connector.

The following is an issue associated with the target system:

An Error Occurs While Disabling a GoToMeeting Account

If you try to disable a GoToMeeting account from Oracle Identity Manager for the first time, the disable task is rejected and the following message appears in the resource history of the account:

```
Disable operation not supported by target.
```

If you try to disable the same account more than once, then an error occurs with a message stating that duplicate schedule item for this task does not allow multiple entries. This is because the **Allow Multiple Instances** check box is not selected for the **Disable User** process task.

Workaround:

You can resolve this error from the Process Definition form of Oracle Identity Manager Design Console. To do so, perform the following steps:

1. Expand **Process Management** and double-click **Process Definition**.
2. Search for and open the **GoToMeeting User** process definition.
3. On the Tasks tab, double-click the row heading of the **Disable User** process task.
4. On the General tab of the Editing Task: Disable User dialog box, select the **Allow Multiple Instances** check box.
5. Click the Save icon.

A

Files and Directories on the GoToMeeting Connector Installation Media

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

> **Note:**
>
> If you have integrated Oracle Identity Manager with GoToMeeting using the OIM AD connector, see Files and Directories On the Installation Media in *Oracle Identity Manager Connector Guide for Microsoft Active Directory User Management* for a list of the files and directories on the OIM AD connector installation media.

Table A-1 lists the files and directories on the installation media of the GoToMeeting connector.

Table A-1 Files and Directories on the Installation Media of the GoToMeeting Connector

File in the Installation Media Directory	Description
bundle/ org.identityconnectors.genericrest-1.0.1115.jar	This JAR is the ICF connector bundle.
configuration/GoToMeeting-CI-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/GoToMeeting-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