Oracle® Cloud

Using the Oracle HCM Cloud Adapter with Oracle Integration 3

F45583-20
January 2026

Oracle Cloud Using the Oracle HCM Cloud Adapter with Oracle Integration 3,

F45583-20

Primary Author: Oracle Corporation

Contents

3 Add the Oracle HCM Cloud Adapter Connection to an Integration

4 Implement Common Patterns Using the Oracle HCM Cloud Adapter

5 Troubleshoot the Oracle HCM Cloud Adapter

About This Content

This guide describes how to configure this adapter as a connection in an integration in Oracle Integration.

Audience

This guide is intended for developers who want to use this adapter in integrations in Oracle Integration.

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 Resources

See these Oracle resources:

- Oracle Cloud at `http://cloud.oracle.com`
- *Using Integrations in Oracle Integration 3*
- *Using the Oracle Mapper with Oracle Integration 3*
- Oracle Integration documentation on the Oracle Help Center.

Conventions

The following text conventions are used in this document.

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

1

Understand the Oracle HCM Cloud Adapter

Review the following conceptual topics to learn about the Oracle HCM Cloud Adapter and how to use it as a connection in integrations in Oracle Integration. A typical workflow of adapter and integration tasks is also provided.

Topics:

Oracle HCM Cloud Adapter Capabilities

The Oracle HCM Cloud Adapter enables you to create an integration with Oracle Human Capital Management (HCM) Cloud applications. You select business objects that an integration receives from Oracle HCM Cloud as a request and as a response.

The Oracle HCM Cloud Adapter enables customers to easily integrate their on-premises or SaaS applications with Oracle HCM Cloud without having to know about the specific details involved in the integration.

The Oracle HCM Cloud Adapter provides the following benefits:

- Supports connecting to private resources that are in your virtual cloud network (VCN) private subnet with a private endpoint. See Connect to Private Resources in *Provisioning and Administering Oracle Integration 3* and Configure the Endpoint Access Type. This type of connection does not use the connectivity agent.
- Integrates easily with the Oracle HCM Cloud application's WSDL file to produce a simplified, integration-centric WSDL.
- Generates automatic mapping to the exposed business object that you select during adapter configuration as a trigger connection. A business object represents a self-contained business document that can be acted upon by the integration. An integration can send requests to create a new record for that business object. They can send a request either to update or delete an existing record for a business object. Integrations can also send requests to retrieve information about one or more records representing that business object.
- Supports invoking of business objects using the HCM Data Loader, regardless of how they were created. The HCM Data Loader is a tool for bulk-loading and maintaining data. The data can be from any source. You use the HCM Data Loader for data migration, ongoing maintenance of Oracle HCM Cloud data, and coexistence scenarios where core HR data is uploaded regularly.
  See Overview of HCM Data Loader and Overview of Automating Data Loading in *HCM Data Loader*.

- Exposes the Business Resource (REST) API, which represents an Oracle Fusion Applications REST API resource.

  You can select parent business resources and their corresponding child business resources on the Operations page in the Adapter Endpoint Configuration Wizard. Support is provided in the invoke (outbound) direction. If you select a top-level resource on the Operations page, you can also select sub-resources on the Sub-Resources page. See Invoke Child Resources Page
- Simplified connection creation: Automatically identifies the required service catalog service WSDL and optional interface catalog URL to use based on the Oracle HCM Cloud host name you specify when creating a new connection on the Connections page.
- Supports consuming extensible flexfields (EFFs) and descriptive flexfields (DFFs) for REST resources. You can select specific EFFs and DFFs in the Adapter Endpoint Configuration Wizard of an Oracle HCM Cloud Adapter invoke connection. You can then map the EFFs and DFFs in the mapper. See Select Extensible and Descriptive Flexfields in an Integration.
- Supports consumption of the Oracle HCM Cloud REST API. This enables the Oracle HCM Cloud Adapter to consume REST services under Oracle HCM Cloud when configured as an invoke connection. See About the REST APIs in *REST API for Oracle Fusion Cloud HCM* for details about supported REST resources.
- Automatically handles security policy details required to connect to the Oracle HCM Cloud application.
- Supports the following security policies for selection during Oracle HCM Cloud Adapter connection configuration:
  - Username Password Token With PGP Key Support
  - Username Password Token
  - OAuth Authorization Code Credentials
  - OAuth using JWT User Assertion
- Provides standard error handling capabilities.
- Enables you to map business objects that have polymorphic data structures.
- Dynamically invokes a REST endpoint/URL at runtime without requiring you to configure any extra invoke connection or REST outbound details. See Invoke an Endpoint Dynamically.
- Supports subscribing to the HCM Atom feed. Atom feeds enable you to track changes made to feed-enabled resources in Oracle Global Human Resources Cloud. For any updates of interest to downstream applications such as new hires, terminations, employee transfers, and promotions, Oracle Global Human Resources Cloud publishes Atom feeds. The Oracle HCM Cloud Adapter's Atom feed subscription feature enables you to select a feed of interest. This feature must be used in a scheduled manner because there is a need to poll for updates at regular intervals. This is done by selecting a scheduled orchestration template when creating the integration.

  This feature is supported when using the Oracle HCM Cloud Adapter as an invoke connection in an integration.

  Prior to HCM release 18*c*, future-dated Atom entries appeared in their respective Atom feed immediately once the future-dated action was taken.

  Starting with HCM release 18*c*, there is a change in the way future-dated updates are returned by the HCM service. Atom feed entries appear by default once the future-dated actions become effective. You can select to process these future-dated Atom entries

immediately on the Operations page of the Adapter Endpoint Configuration Wizard. The default behavior is to process future-dated Atom updates in the future.

See Poll in *REST API for Oracle Fusion Cloud HCM*.

See Atom Feeds in *REST API for Oracle Fusion Cloud HCM*.

- Enables you to upload files to Oracle WebCenter Content (Universal Content Manager) in encrypted or unencrypted format. Oracle WebCenter Content provides a unified repository to store unstructured content, enabling organizations to deliver the content to business users in the proper format. See Upload a File to Oracle WebCenter Content.

  > **Note**
  >
  > Downloading files from Oracle WebCenter Content is not supported.

- Supports HCM data extracts, a flexible tool for generating data files and reports. The Oracle HCM Cloud Adapter works as an extract discovering tool under Oracle HCM Cloud. The data extract process is automated as per the following steps:
  - Invoking the given HCM data extract by the client outside the Oracle HCM Cloud Adapter.
  - Discovering the extract as delivered in Oracle WebCenter Content.
  - Fetching the extract output in its format as it is delivered to Oracle WebCenter Content and persisting the extract to Oracle Integration staging.
  - Defining the XML schema by the user for the extract output for transformation using a stage file action that is available in an orchestrated integration.

    > **Note**
    >
    > The user must understand how to download the schema and how to schedule and run HCM extracts from the Oracle HCM Cloud user interface. See Define Extracts in *HCM Extracts*.

  This feature is supported when using the Oracle HCM Cloud Adapter as an invoke connection in an integration.

- Supports a set of SOAP services. For the supported list, see Business Object Services in *SOAP Web Services for HCM*. The following integration behavior occurs:
  - For new integrations, only the supported SOAP services are displayed for selection when configuring the Oracle HCM Cloud Adapter in the Adapter Endpoint Configuration Wizard. Unsupported SOAP services are not displayed.
  - For existing integrations, you can edit, view, and activate old integration as before. If you add a new adapter endpoint to an existing integration, only supported SOAP services are available for selection in the Adapter Endpoint Configuration Wizard.
- Supports MTOM attachments by default for the ErpIntegrationService, ErpObjectAttachmentService, and GenericSOAPService services for new connections starting with release 25.02. If the response from these services includes an attachment, they are received as an attachment reference instead of base64 content. To process the response, the attachment reference can be directly used by a stage file action to read. This eliminates the need for using mapper functions or writing to a stage file action explicitly to get the reference from base64. This behavior speeds up the development and execution of integrations.

- Supports JWT user assertions with the JWT User Assertion security policy. JWT assertions enable you to invoke a service provider that does not regard an OAuth client secret as secure. Trust is established with a key pair exchange instead of a client secret. See Configure Connection Security and Propagate OAuth User Identity Between Services.

Oracle HCM Cloud Adapter Restrictions

Note the following Oracle HCM Cloud Adapter restrictions.

- The System for Cross-Domain Identity Management (SCIM) REST API under Oracle HCM Cloud is not discoverable through the Oracle Application Development Framework. Therefore, the Oracle HCM Cloud Adapter is unable to list those resources during design time. As an alternative, use the REST Adapter with the SCIM REST API under Oracle HCM Cloud.
- Downloading files from Oracle WebCenter Content is not supported.
- Oracle Applications Cloud REST API framework versions 6 and above are not supported. Using these versions results in a runtime error. See Set the REST Framework Version in *REST API for Oracle Fusion Cloud HCM*.

> **Note**
>
> There are overall service limits with Oracle Integration. A service limit is the quota or allowance set on a resource. See Service Limits.

What Application Version Is Supported?

For information about which application version is supported by this adapter, see the Connectivity Certification Matrix.

Oracle HCM Cloud Adapter Use Cases

Common use cases for the Oracle HCM Cloud Adapter are as follows:

- New employee onboarding
- Integrate with third-party benefits providers such as health insurance providers
- Extract employee data monthly from Oracle HCM Cloud for payroll

Workflow to Create and Add an Oracle HCM Cloud Adapter Connection to an Integration

You follow a very simple workflow to create a connection with an adapter and include the connection in an integration in Oracle Integration.

Step	Description	More Information
1	Decide where to work	• Work in a project (see why working with projects is preferred in *Using Integrations in Oracle Integration 3*). • Work outside a project.

Step	Description	More Information
2	Create the adapter connections for the applications you want to integrate. The connections can be reused in multiple integrations and are typically created by the administrator.	Create an Oracle HCM Cloud Adapter Connection
3	Create the integration. When you do this, you add trigger and invoke connections to the integration.	Understand Integration Creation and Best Practices and Add the Oracle HCM Cloud Adapter Connection to an Integration
4	Map data between the trigger connection data structure and the invoke connection data structure.	Map Data in *Using Integrations in Oracle Integration 3*
5	(Optional) Create lookups that map the different values used by those applications to identify the same type of object (such as gender codes or country codes).	Manage Lookups in *Using Integrations in Oracle Integration 3*
6	Activate the integration.	Manage Integrations in *Using Integrations in Oracle Integration 3*
7	Monitor the integration on the dashboard.	Monitor Integrations During Runtime in *Using Integrations in Oracle Integration 3*
8	Track payload fields in messages during runtime.	Assign Business Identifiers for Tracking Fields in Messages and Track Integration Instances in *Using Integrations in Oracle Integration 3*
9	Manage errors at the integration level, connection level, or specific integration instance level.	Manage Errors in *Using Integrations in Oracle Integration 3*

2

Create an Oracle HCM Cloud Adapter Connection

A connection is based on an adapter. You define connections to the specific cloud applications that you want to integrate.

Topics:

Prerequisites for Creating a Connection

Satisfy the following prerequisites specific to your environment to create a connection with the Oracle HCM Cloud Adapter.

Subscribe to Oracle HCM Cloud

Subscribe to Oracle HCM Cloud. This action enables you to create an Oracle HCM Cloud user account with the correct privileges. You specify this user account when creating an Oracle HCM Cloud Adapter connection on the Connections page.

For information about specifying these credentials on the Connections page, see Configure Connection Security. For information about subscribing, see Oracle HCM Cloud.

Request Access to Atom Feeds and REST APIs Capabilities on Your Oracle HCM Cloud Instances

Oracle HCM Cloud Atom feeds and Oracle HCM Cloud REST APIs support is available by default in the Oracle HCM Cloud Adapter in Oracle Integration. However, for the Oracle HCM

Cloud application, you must first request access to Atom feeds and REST APIs capabilities on your Oracle HCM Cloud instance(s).

See the steps described in My Oracle Support Note 2060899.1. Once these are enabled in your Oracle HCM Cloud application, Atom feeds and REST APIs appear in the Oracle HCM Cloud Adapter in Oracle Integration.

Assign Required Roles to an Integration User

To use the Oracle HCM Cloud Adapter in an integration, you must assign specific roles to an integration user.

Associating the Integration User with the Following Roles and Privileges

You associate the user with the following roles and privileges.

Role	Description
ALL_INTEGRATION_POINTS_ALL_DATA	Starting with release 12, this role is no longer supported. When existing customers upgrade to release 12, users with this role continue using it, although it is hidden from the Security Console. If you create a new integration user in release 12 or later, you cannot assign this role.
ORA_HRC_HUMAN_CAPITAL_MANAGEMENT_INTEGRATION_SPECIALIST_JOB	Human Capital Management Integration Specialist. The role applies to Releases 12 and 13.
AttachmentsUser	Provides access to the Attachments security group to download the log file or the output file with the HCM Integration Service. Starting with Release 12, this role is automatically shipped. You must verify that this role is automatically assigned to the user.
SOAOperator	The SOA Operator role.
FND_MANAGE_CATALOG_SERVICE_PRIV	Role for managing the web services catalog.
For Oracle CRM Cloud implementations, you can also assign the Customer Relationship Management Application Administrator role.	See Customer Relationship Management Application Administrator (Job Role) of *Security Reference for CX Sales and B2B Service*.

Additional roles may be required as per each interface requirements.

Using the Security Console

Use the Security Console to manage application security such as roles, users, certificates, and administration tasks. Access to the Security Console is provided by the predefined **Security Manager** role. Access the Security Console in the following ways:

- Use the Manage Job Roles or Manage Duties tasks in the Setup and Maintenance work area.
- Select **Navigator** > **Tools** > **Security Console**.

Create an HCM-compliant .dat File to Use the HCM Data Loader

If you want to use the HCM Data Loader for bulk-loading and maintaining data, you must create a `.dat` file that is HCM-compliant. See your Oracle HCM Cloud documentation for instructions.

Upload Files to Oracle WebCenter Content

You must satisfy the following prerequisites if you want to upload a file to Oracle WebCenter Content (Universal Content Manager) with the Oracle HCM Cloud Adapter.

- Create a PGP Public Key for Encrypted File Upload:

  To upload encrypted files, a PGP public key is required. You must generate the PGP public key and save it for upload. The supported algorithm for the public key is RSA for encryption and the key size must be 1024 bits in length.

  The process for uploading files into Oracle HCM Cloud is:

  – You encrypt files using the Oracle HCM Cloud public key.

  – The data-loading process decrypts files using the Oracle HCM Cloud private key.

  See Set up Encryption for File Transfer in *HCM Data Loader*.

- Configure Security and User Access

  Once you have configured security groups and doc accounts for the file to upload, you can configure the Oracle HCM Cloud Adapter to upload the file to Oracle WebCenter Content.

Perform Prerequisites to Set Up the OAuth Authorization Code Credentials Security Policy

Perform the following prerequisites to set up the OAuth Authorization Code Credentials security policy with an Oracle Fusion Applications identity domain or a non-Oracle Fusion Applications identity domain (for example, the Oracle Integration identity domain).

Topics:

- Set Up the OAuth Authorization Code Credentials Security Policy with the Oracle Fusion Applications Identity Domain
- Set Up the OAuth Authorization Code Credentials Security Policy with a Non-Oracle Fusion Applications Identity Domain

> **Note**
>
> **The use of a non-Oracle Fusion Applications identity domain is being retired.**
>
> – Create all new connections with the OAuth Authorization Code Credentials security policy in an Oracle Fusion Applications identity domain. See Set Up the OAuth Authorization Code Credentials Security Policy with the Oracle Fusion Applications Identity Domain.
>
> – Existing customers who use a non-Oracle Fusion Applications identity domain (for example, an Oracle Integration identity domain) are being scheduled for migration to an Oracle Fusion Applications identity domain with a completion date of sometime this year. See Identity Upgrade Overview.
>
> After migration, you must reconfigure the following:
>
> * Reconfigure your OAuth resource settings in the Oracle Cloud Console to point to the Oracle Fusion Applications identity domain. See Set Up the OAuth Authorization Code Credentials Security Policy with the Oracle Fusion Applications Identity Domain.
> * Reconfigure your OAuth Authorization Code Credentials security policy connections (for example, the client ID, client secret, authorization code URI, and access token URI). Ensure that you provide consent and test the connection. See Configure Connection Security.
> * Reactivate the integrations using the updated connection. See Reactivate Integrations after a Connection Update.

Set Up the OAuth Authorization Code Credentials Security Policy with the Oracle Fusion Applications Identity Domain

You must create a resource application to represent the Oracle Fusion Applications resource and a client application for Oracle Integration to use the OAuth Authorization Code Credentials security policy. Once these tasks are completed, you can successfully configure a connection on the Connections page. You do not need to create any JWT signing certificates for upload into Oracle Fusion Applications.

- Create an Identity Domain Resource Application to Represent the Oracle Fusion Applications Resource
- Create the Confidential Client Application for Oracle Integration
- Resolve Errors That Occur When Clicking Provide Consent

Create an Identity Domain Resource Application to Represent the Oracle Fusion Applications Resource

1. Create an identity domain resource application to represent the Oracle Fusion Applications resource.
   a. Log in to the identity domain as the domain administrator.
   b. In the menu bar, click **Identity & Security**.
   c. Click **Domains**.
   d. Select your compartment.
   e. Click the identity domain.

f. In the menu bar, click **Integrated applications**.
This is the location at which you create the client application for your grant type.

g. Click **Add application**.

h. Select **Confidential Application**, then click **Launch workflow**.

2. a. Provide a name (for example, `FA Resource`), and click **Submit**.

b. Click the **OAuth configuration** tab, then the **Edit OAuth configuration** subtab.

c. In the **Resource server configuration** section, select **Configure this application as a resource server now**.

d. (Optional) In the **Configure application APIs that need to be OAuth protected** section, select a value from the **Access token expiration (seconds)** list.

e. Click the **Allow token refresh** toggle.

f. In the **Refresh token expiration** (seconds) list, select a value.

g. In the **Primary audience** field, add the Oracle Fusion Applications URL and port. This is the primary recipient where the token is processed.

`https://FA_URL:443`

h. Click the **Add scope** toggle, then click **Add**.

i. In the **Scope** field, enter `/`.

j. In the **Description** field, enter **All**.

k. Select **Requires user consent**.

l. Click **Add**, then click **Submit**.

m. From the **Actions** menu at the top, select **Activate**, and then **Activate application** to activate the application for use. The resource server representing the resource is now active.

Create the Confidential Client Application for Oracle Integration

1. Sign in as the identity domain administrator to the Oracle Cloud Console.
2. In the menu bar, click **Identity & Security**.
3. Click **Domains**.
4. Select your compartment.
5. Click the identity domain.
6. In the menu bar, click **Integrated applications**.
7. Click **Add application**.
8. Select **Confidential Application**, then click **Launch workflow**.
9. Enter a name. The remaining fields on this page are optional and can be ignored.
10. Click **Submit**.
11. Click the **OAuth configuration** tab, then the **Edit OAuth configuration** subtab.

12. In the **Client configuration** panel, select **Configure this application as a client now**.
13. For the authorization code, select **Refresh token** and **Authorization code** in the **Allowed grant types** section.
14. In the **Redirect URL** field, enter the redirect URL of the client application. After user login, this URL is redirected to with the authorization code. You can specify multiple redirect URLs. This is useful for development environments in which you have multiple instances, but only one client application due to licensing issues. For example:

> **Note**
>
> If you don't know the following information, check with your administrator:
>
> - If your instance is new or upgraded from Oracle Integration Generation 2 to Oracle Integration 3.
> - The complete instance URL with the region included (required for new instances).

For Connections…	Include the Region as Part of the Redirect URL?	Example of Redirect URL to Specify…
Created on new Oracle Integration 3 instances	Yes.	`https://`*`OIC_instance_URL`*`.`*`region`*`.ocp.oraclecloud.com/icsapis/agent/oauth/callback`
Created on instances upgraded from Oracle Integration Generation 2 to Oracle Integration 3	No. This applies to both: • New connections created after the upgrade • Existing connections that were part of the upgrade	`https://`*`OIC_instance_URL`*`.ocp.oraclecloud.com/icsapis/agent/oauth/callback`

For the OAuth authorization code to work, the redirect URI must be set properly.

15. Click the **Add Resources** toggle.
16. Click **Add scope** to add appropriate scopes.
If the Oracle Fusion Applications instance is federated with the identity domain, the Oracle Integration cloud service application is listed among the resources for selection. This enables the client application to access Oracle Integration.
17. Search for the Oracle Fusion Applications resource application created in Create an Identity Domain Resource Application to Represent the Oracle Fusion Applications Resource.
18. Find and expand the resource.
19. Select the scope, then click **Add**.
20. Click **Submit**.

The Details page shows the client ID and client secret values in the **General Information** section.

21. Copy and save these values. You need this information when creating a connection for the OAuth Authorization Code Credentials security policy on the Connections page.
Note the following details for successfully authenticating your account on the Connections page.

If The...	Then...
Identity domain safeguarding Oracle Integration and the Oracle Fusion Applications resource application are the same.	Log in to Oracle Integration using the local Oracle Fusion Applications user created earlier. You must create a connection and click **Provide Consent** on the Connections page for authentication to succeed.
Identity domain safeguarding Oracle Integration and the Oracle Fusion Applications resource application are different.	Log in to Oracle Integration using a general Oracle Integration developer account, create a connection, and click **Provide Consent** on the Connections page. You need to log in to the Oracle Fusion Applications resource identity domain application using the local Oracle Fusion Applications user account created earlier.

22. From the **Actions** menu at the top, select **Activate**, and then **Activate application** to activate the client application for use.

Resolve Errors That Occur When Clicking Provide Consent

After you configure the OAuth Authorization Code Credentials security policy on the Connections page, you must test your connection.

If you are logged in to Oracle Integration with an Oracle Integration user account and click **Provide Consent** to test the OAuth flow, consent is successful. However, when you test the connection, it fails with an `Unauthorized 401` error.

This error occurs because the Oracle Integration user account with which you logged in is not part of Oracle Fusion Applications.

1. Log out of Oracle Integration and log back in with a user account that exists in Oracle Fusion Applications.
2. Return to the Connections page and retest the connection.
The connection is successful this time.

Set Up the OAuth Authorization Code Credentials Security Policy with a Non-Oracle Fusion Applications Identity Domain

You must set up trust between Oracle Fusion Applications and an identity domain and create a client application for Oracle Integration to use the OAuth Authorization Code Credentials security policy. Once these tasks are completed, you can successfully configure a connection

on the Connections page. Use this option when you are integrating with a non-Oracle Fusion Applications identity domain, such as the Oracle Integration identity domain.

> **Note**
>
> **The use of a non-Oracle Fusion Applications identity domain is being retired**. Customers using this identity domain are being migrated. See Identity Upgrade Overview.

- Set Up Trust Between Oracle Fusion Applications and an Identity Domain
- (Optional) Create a Local User
- Create the Confidential Client Application for Oracle Integration
- Avoid Potential Errors When Testing Your Connection with a Nonfederated User Account

Set Up Trust Between Oracle Fusion Applications and an Identity Domain

1. Get the JWK signing certificates from the identity domain of Oracle Integration.
   a. Get the REST API of the identity domain endpoint that gives you the signing certificate endpoint. For example:

      ```
      /admin/v1/SigningCert/jwk
      ```

      See Getting Started with the Identity Domains REST API.
   b. Copy the endpoint.
   c. Get the identity domain URL from the Oracle Cloud Console or from the Oracle Integration **About** menu.
   d. Add that URL to the front of the signing certificate and use a tool (for example, `postman`) to invoke the REST APIs. For example:

      ```
      https://identity_domain_URL.identity.oraclecloud.com/admin/v1/
      SigningCert/jwk
      ```

   e. Perform a GET call to retrieve the payload of the signing keys. There are two sections in the payload:
      - Identity domain signing key
      - Certificate authority (CA) signing key

      Examples of the type of response you receive are provided. See Retrieve the Tenant's Signing Certificate in JWK Format.
   f. Copy both signing key sections into separate files. Note that the headers and footers in the files must be in the following exact format to be successfully uploaded to Oracle Fusion Applications:

      ```
      -----BEGIN CERTIFICATE-----
       content_of_signing_key
      . . .
      . . .
      -----END CERTIFICATE-----
      ```

You can validate the content. For example:

```
openssl x509 -in identity.cert -noout -text
```

2. File a service request (SR) with Oracle Fusion Applications Support that includes the following details:
   - **SR Summary**: Set Up Trust Between Oracle Fusion Applications and OCI Identity Domain
   - **Category**: Login, Logout and SSO

   Attach your certificates for upload. You cannot upload the certificates yourself.
3. Create a resource application in an Oracle Integration identity domain to represent the Oracle Fusion Applications resource.
   a. Log in to the identity domain as the domain administrator.
   b. In the navigation pane, click **Identity & Security**.
   c. Click **Domains**.
   d. Select your compartment.
   e. Click the identity domain.
   f. In the navigation pane, click **Integrated applications**.
   g. Click **Add application**.
   h. Select **Confidential Application**, then click **Launch workflow**.
   i. Provide a name (for example, `FA Resource`), and click **Submit**.
   j. Click the **OAuth configuration** tab, then the **Edit OAuth configuration** subtab.
   k. In the **Resource server configuration** section, click **Configure this application as a resource server now**.
   l. (Optional) In the **Configure application APIs that need to be OAuth protected** section, select a value from the **Access token expiration (seconds)** list.
   m. Click the **Allow token refresh** toggle.
   n. In the **Refresh token expiration** (seconds) list, select a value.
   o. In the **Primary Audience** field, add the Oracle Fusion Applications URL and port. This is the primary recipient where the token is processed.

      ```
      https://FA_URL:443
      ```

   p. Click the **Add scope** toggle, then click **Add**.
   q. In the **Scope** field, enter `/`.
   r. In the **Description** field, enter **All**.
   s. Select **Requires user consent**.
   t. Click **Add**, then click **Submit**.
   u. From the **Actions** menu at the top, select **Activate**, and then **Activate application** to activate the client application for use.

(Optional) Create a Local User

> **Note**
>
> The following step is required if the Oracle Fusion Applications user is *not* federated with an identity domain or whichever identity provider you are using.

1. Create an identity domain local user. *Carefully* review the following table to see if you already have a local user.

Scenario	Do I Need to Create a Local User?
You have an Oracle Fusion Applications user federated with the identity domain that is protecting Oracle Integration.	No. You do not need to create the local identity domain Oracle Fusion Applications user. This is because identity domain already has Oracle Fusion Applications users in its repository.
You do *not* have federation between Oracle Fusion Applications and the identity domain that is protecting Oracle Integration.	Yes. You must create the local identity domain Oracle Fusion Applications user that you plan to use with the OAuth setup in Oracle Integration.

The identity domain administrator must create a nonfederated local username in the identity domain that matches the user in Oracle Fusion Applications. If you have already used and invoked Oracle Fusion Applications REST endpoints, you likely already created a user with the necessary roles and accesses to invoke the REST endpoints of Oracle Fusion Applications. This user must be created in the identity domain and have a local user password.

Create the Confidential Client Application for Oracle Integration

1. Sign in as the identity domain administrator to the Oracle Cloud Console.
2. In the navigation pane, click **Identity & Security**.
3. Click **Domains**.
4. Select your compartment.
5. Click the identity domain.
6. In the navigation pane, click **Integrated applications**.
7. Click **Add application**.
8. Select **Confidential Application**, then click **Launch workflow**.
9. Enter a name. The remaining fields on this page are optional and can be ignored.
10. Click **Submit**.
11. Click the **OAuth configuration** tab, then the **Edit OAuth configuration** subtab.
12. In the **Client configuration** box, select **Configure this application as a client now**.
13. For the authorization code, select **Refresh token** and **Authorization code** in the **Allowed grant types** section.
14. In the **Redirect URL** field, enter the redirect URL of the client application. After user login, this URL is redirected to with the authorization code. You can specify multiple redirect

URLs. This is useful for development environments in which you have multiple instances, but only one client application due to licensing issues. For example:

> **Note**
>
> If you don't know the following information, check with your administrator:
>
> - If your instance is new or upgraded from Oracle Integration Generation 2 to Oracle Integration 3.
> - The complete instance URL with the region included (required for new instances).

For Connections…	Include the Region as Part of the Redirect URL?	Example of Redirect URL to Specify…
Created on new Oracle Integration 3 instances	Yes.	`https://`*`OIC_instance_URL`*`.`*`region`*`.ocp.oraclecloud.com/icsapis/agent/oauth/callback`
Created on instances upgraded from Oracle Integration Generation 2 to Oracle Integration 3	No. This applies to both: • New connections created after the upgrade • Existing connections that were part of the upgrade	`https://`*`OIC_instance_URL`*`.ocp.oraclecloud.com/icsapis/agent/oauth/callback`

For the OAuth authorization code to work, the redirect URI must be set properly.

15. Click the **Add scope** toggle, then click **Add**.
    If the Oracle Fusion Applications instance is federated with the identity domain, the Oracle Integration cloud service application is listed among the resources for selection. This enables the client application to access Oracle Integration.
16. Search for the Oracle Fusion Applications resource application created in Set Up Trust Between Oracle Fusion Applications and an Identity Domain.
17. Find and expand the resource.
18. Select the scope, then click **Add**.
19. Click **Submit**.
    The details page shows the client ID and client secret values.
20. Copy and save these values. You need this information when creating a connection for the OAuth Authorization Code Credentials security policy on the Connections page.
    Note the following details for successfully authenticating your account on the Connections page.

If The...	Then...
Identity domain safeguarding Oracle Integration and the Oracle Fusion Applications resource application are the same.	Log in to Oracle Integration using the local Oracle Fusion Applications user created earlier. You must create a connection and click **Provide Consent** on the Connections page for authentication to succeed.
Identity domain safeguarding Oracle Integration and the Oracle Fusion Applications resource application are different.	Log in to Oracle Integration using a general Oracle Integration developer account, create a connection, and click **Provide Consent** on the Connections page. You need to log in to the Oracle Fusion Applications resource identity domain application using the local Oracle Fusion Applications user account created earlier.

21. From the **Actions** menu at the top, select **Activate**, and then **Activate application** to activate the client application for use.

Avoid Potential Errors When Testing Your Connection with a Nonfederated User Account

After you configure the OAuth Authorization Code Credentials security policy on the Connections page, you must test your connection.

If you are logged in to Oracle Integration with an Oracle Integration user account and click **Provide Consent** to test the OAuth flow, consent is successful. However, when you test the connection, it fails with an `Unauthorized 401` error.

This error occurs because the Oracle Integration user account with which you logged in is not part of Oracle Fusion Applications.

1. Log out of Oracle Integration and log back in with a user account that exists in Oracle Fusion Applications.
2. Return to the Connections page and retest the connection.
   The connection is successful this time.

Verify the Status of Location-Based Access Control (LBAC)

Check if you have enabled Location-Based Access Control (LBAC) for Fusion Applications (for Oracle HCM Cloud).

If LBAC is enabled, you must allowlist (explicitly allow identified entities access) the Oracle Integration NAT Gateway IP address in your LBAC. If you do not perform this task, you can receive a `401 Access Denied` error or `403 Forbidden` error from Oracle Fusion Applications.

See How Location-Based Access Works in *Securing SCM* and Doc ID 2615294.1 at Oracle Support Services.

Perform Prerequisites to Use the JWT User Assertion Security Policy

You must perform prerequisites to use the JWT User Assertion security policy.

Topics:

- Configure a Confidential Application to Use the JWT User Assertion Security Policy
- Configure JWT Assertions for Outbound Use

Configure a Confidential Application to Use the JWT User Assertion Security Policy

You must generate a private key and configure a confidential application to use the JWT User Assertion security policy.

See Prerequisites for JWT User Assertion in *Using the REST Adapter with Oracle Integration 3*.

Configure JWT Assertions for Outbound Use

Perform the following prerequisites to use JWT assertions.

- Take the private key you generated in Configure a Confidential Application to Use the JWT User Assertion Security Policy and upload it on the Certificates page. See Upload a Certificate to Connect with External Services.
  The service provider typically provides instructions on how to generate the signing keys and the format. For an example, see Required Keys and OCIDs.
- Create the JWT header and JWT payload JSON files. You upload both files on the Connections page when configuring the adapter to support JWT assertions.
  Note the following details about the JWT payload JSON file:
  - The `iss`, `exp`, `sub`, and `aud` claims are mandatory. Oracle Integration validates that these claims are present. Anything else you upload depends on the provider you are trying to call. For example, NHS may require additional claims.
  - The `iat` (issued at), `exp` (expiry), `nbf` (not before), and `jti` (JWT ID) claims are dynamically calculated if present in the JWT payload JSON file. If you manually provide values for these claims, they are replaced with dynamically-calculated values.
  - Any remaining claims are optional and depend upon the provider you are calling.

  For example:

JWT Header JSON File Example

```
{
    "alg" : "RS256",
    "typ": "JWT",
    "kid": "fajwt2"
}
```

Where:

- `alg`: The algorithm to use.
- `typ`: A JWT assertion typically set to `JWT`.
- `kid`: A key identifier that is uniquely-generated and associated with the uploaded signing key.

JWT Payload JSON File Example

```
{
    "iss":
"f6c9d437eed64e2a8f2b045e39e2e03f",
    "sub": "admin.user",
    "aud": "https://
identity.oraclecloud.com/",
    "exp": "1739412427"
    "iat": "1727372629"
    "jti": "12345"
  }
```

Where:

- JWT issuer (`iss`): A unique identifier for the entity that issued the assertion. This is typically the entity that holds the key material used to sign or integrity-protect the assertion. Examples of issuers are OAuth clients (when assertions are self-issued) and third-party security token services. If the assertion is self-issued, the issuer value is the client identifier (`client_id`). If the assertion was issued by a security token service (STS), the issuer must identify the STS in a manner recognized by the authorization server. The assertion must contain an issuer.
- JWT subject (`sub`): The subject typically identifies an authorized accessor for which the access token is being requested (that is, the resource owner or an authorized delegate). In some cases, this may be a pseudo anonymous identifier or other value denoting an anonymous user. When the client is acting on behalf of itself, the subject must be the value of the client's `client_id`. The assertion must contain a subject.
- JWT audience (`aud`): A value that identifies the party or parties to process the assertion. The assertion must contain an audience that identifies the authorization server as the intended audience. The authorization server must reject any assertion that does not contain its own identity as the intended audience (in this case, for an Oracle Cloud Infrastructure Identity and Access Management identity domain, `https://identity.oraclecloud.com/`).
- Expires at (`exp`): The time at which the assertion expires. While the serialization may differ by assertion format, the time must be expressed in UTC format with no time zone component. The assertion must contain an expires-at entity that limits the window during which the assertion can be used. The authorization server must reject expired assertions (subject to allowable

JWT Header JSON File Example	JWT Payload JSON File Example
	clock skew between systems). The authorization server may reject assertions with an expires-at attribute value that is unreasonably far in the future. – Issued at (`iat`): The time at which the JWT was issued. – JWT identifier (`jti`): A unique identifier for the JWT. This helps to prevent replay attacks and ensures the token is only used once.

Specifying the Oracle HCM Cloud Service Catalog Service WSDL or Event Catalog URL

You must specify a mandatory Oracle HCM Cloud service catalog service WSDL (for accessing business objects) and optionally an event catalog URL (for accessing event subscriptions).

Obtaining the Oracle HCM Cloud Catalog Service WSDL

WSDL Requirements	Where Do You Get the WSDL
The URL must be that of a service catalog service WSDL. The service catalog service enables clients to retrieve information about all public Oracle Fusion Application service endpoints available for that instance. The information it returns is specific to the particular cloud instance and also reflects the new services that may have been introduced in patches applied to the instance. This service is used to programmatically discover the SOAP services available on the cloud instance and retrieve the necessary metadata to invoke the SOAP services to manage business objects.	The developer creating an Oracle HCM Cloud connection must work with the Oracle HCM Cloud service administrator to get the concrete WSDL URL for the service catalog service provisioned for the specific SaaS application. The concrete WSDL URL must be supplied while creating the connection.

Prerequisites

This section describes how to derive the external virtual host and port for a tokenized service WSDL. The topology information in the Topology Registration setup task contains the external virtual host and port for the domains and applications. The following instructions describe the steps for deriving the values using the service catalog service WSDL URL as an example: `https://atf_server:port/fndAppCoreServices/ServiceCatalogService`.

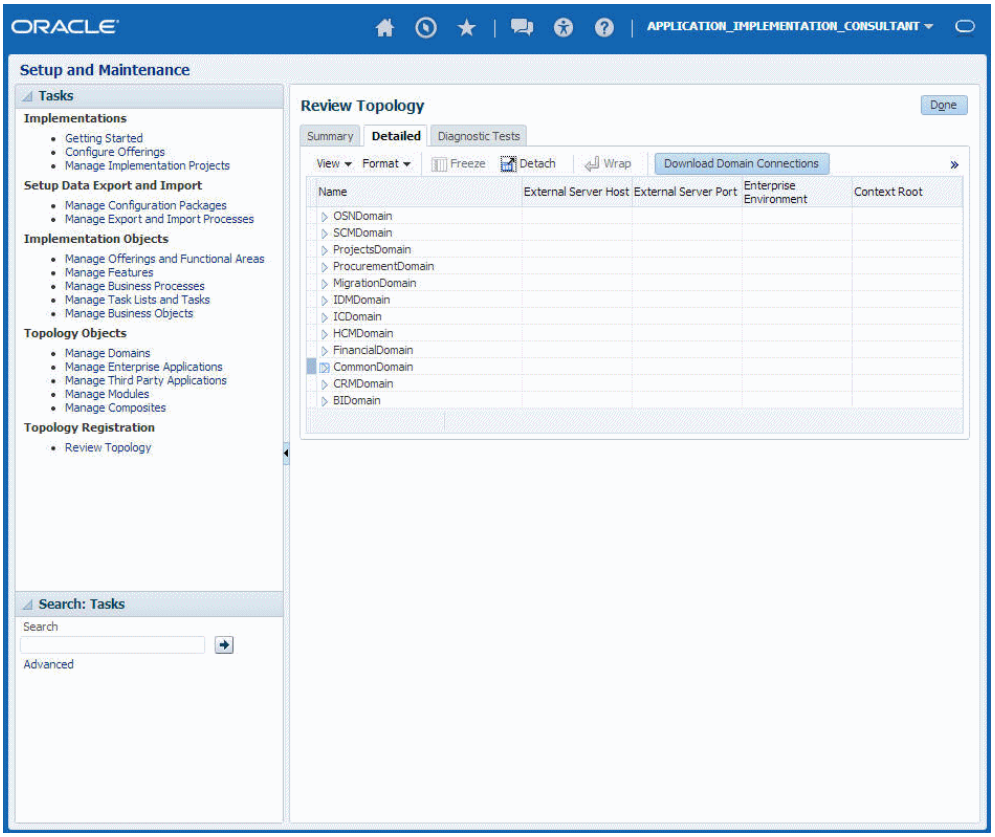
To access the Review Topology page, the ASM_REVIEW_TOPOLOGY_HIERARCHY_PRIV entitlement must be granted to the user's job role. The entitlement is granted to the ASM_APPLICATION_DEPLOYER_DUTY duty role, which is inherited by the duty roles ASM_APPLICATION_DEVELOPER_DUTY and ASM_APPLICATION_ADMIN_DUTY.

If the menu items and tasks described in the following procedure are not available in your cloud instance, your user account is missing the required role. Contact your cloud instance security administrator for assistance.

1. Log in to the cloud instance.
2. Click the **Navigator** icon in the global area in the top part of the window, then chose **Setup and Maintenance** under the **Tools** heading.

3. Select **Review Topology** under the **Topology Registration** section in the **Tasks** regional area on the left side of the window.
4. Click the **Detailed** tab in the middle of the window.

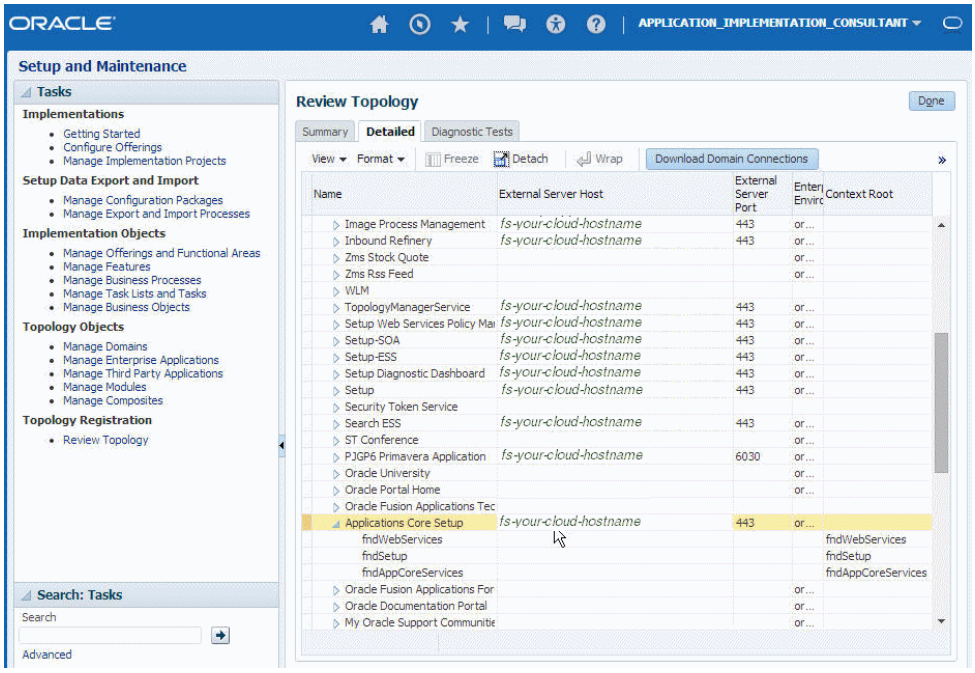
   The tab shows the list of domains configured in the cloud instance.



5. Map the token name for the service path value to the domain name in the Topology Manager:

Token Name in Service Path	Domain Name
atf_server	CommonDomain
crm_server	CRMDomain
fin_server	FinancialDomain
hcm_server	HCMDomain
ic_server	ICDomain
prc_server	ProcurementDomain
prj_server	ProjectsDomain
scm_server	SCMDomain

6. Expand the domain name and select any external virtual host and port for the J2EE applications that are deployed on the domain. In the sample window, the values for this particular instance are **fs-your-cloud-hostname** and **443**, respectively.



7. Replace the *domainName_server*:*PortNumber* with the external virtual host and port identified in the previous step. For example:

   ```
   https://fs-your-cloud-hostname:port/fndAppCoreServices/ServiceCatalogService?wsdl
   ```

Obtaining the Event Catalog URL

You must know the CRM URL format to access the CRM application user interface. Follow the URL format to determine the event catalog URL. For example, if the CRM URL format is:

```
https://fusxxxx-crm-ext.us.oracle.com/customer/faces/CrmFusionHome
```

Then the event catalog URL is:

```
https://fusxxxx-crm-ext.us.oracle.com/soa-infra
```

For Fusion Applications Releases 10 Through 12

Obtain the Oracle Fusion Applications Releases 10 through 12 service catalog service WSDLs and interface catalog URLs through the following methods.

- Obtain the Service Catalog Service WSDL for Releases 10 Through 11
- Obtain the Service Catalog Service WSDL for Release 12
- Obtain the Interface Catalog URL

Obtain the Service Catalog Service WSDL for Releases 10 Through 11

WSDL Requirements	Where Do You Get the WSDL?
The URL must be that of a service catalog service WSDL. The service catalog service is a Fusion Application service that returns a list of external services available for integration. It allows clients to retrieve information about all public Fusion Application service endpoints available for that instance. The service catalog service enables clients to retrieve information about all public Oracle Fusion Application service endpoints available for that instance. The information it returns is specific to the particular cloud instance and also reflects the new services that may have been introduced in patches applied to the instance. This service is used to programmatically discover the SOAP services available on the cloud instance and retrieve the necessary metadata to invoke the SOAP services to manage business objects.	The developer creating an Oracle HCM Cloud connection must work with the Oracle HCM Cloud service administrator to get the concrete WSDL URL for the service catalog service provisioned for the specific SaaS application.

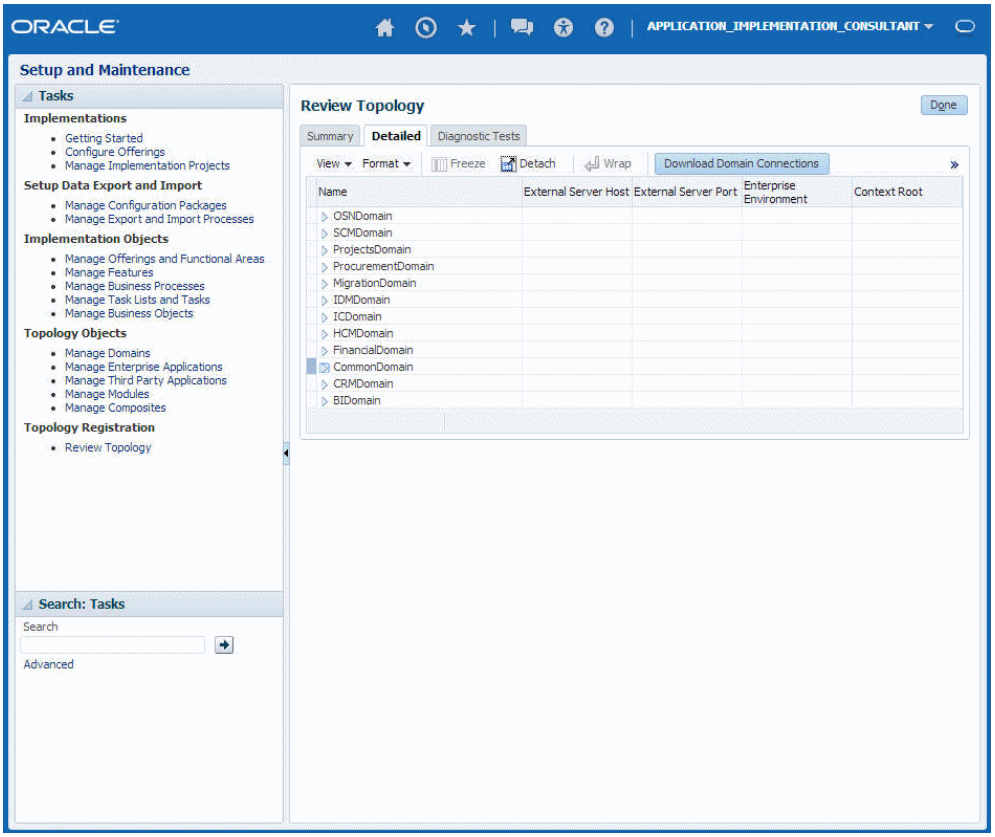
This section describes how to derive the external virtual host and port for a tokenized service catalog service WSDL. The topology information in the Topology Registration setup task contains the external virtual host and port for the domains and applications. The following instructions describe the steps for deriving the values using the service catalog service WSDL URL as an example: `https://atf_server:port/fndAppCoreServices/ServiceCatalogService`.

To access the Review Topology page, the ASM_REVIEW_TOPOLOGY_HIERARCHY_PRIV entitlement must be granted to the user's job role. The entitlement is granted to the ASM_APPLICATION_DEPLOYER_DUTY duty role, which is inherited by the duty roles ASM_APPLICATION_DEVELOPER_DUTY and ASM_APPLICATION_ADMIN_DUTY.

If the menu items and tasks described in the following procedure are not available in your cloud instance, your user account is missing the required role. Contact your cloud instance security administrator for assistance.

1. Log in to the cloud instance.
2. Click the **Navigator** icon in the global area in the top part of the window, then chose **Setup and Maintenance** under the **Tools** heading.
3. Select **Review Topology** under the **Topology Registration** section in the **Tasks** regional area on the left side of the window.
4. Click the **Detailed** tab in the middle of the window.

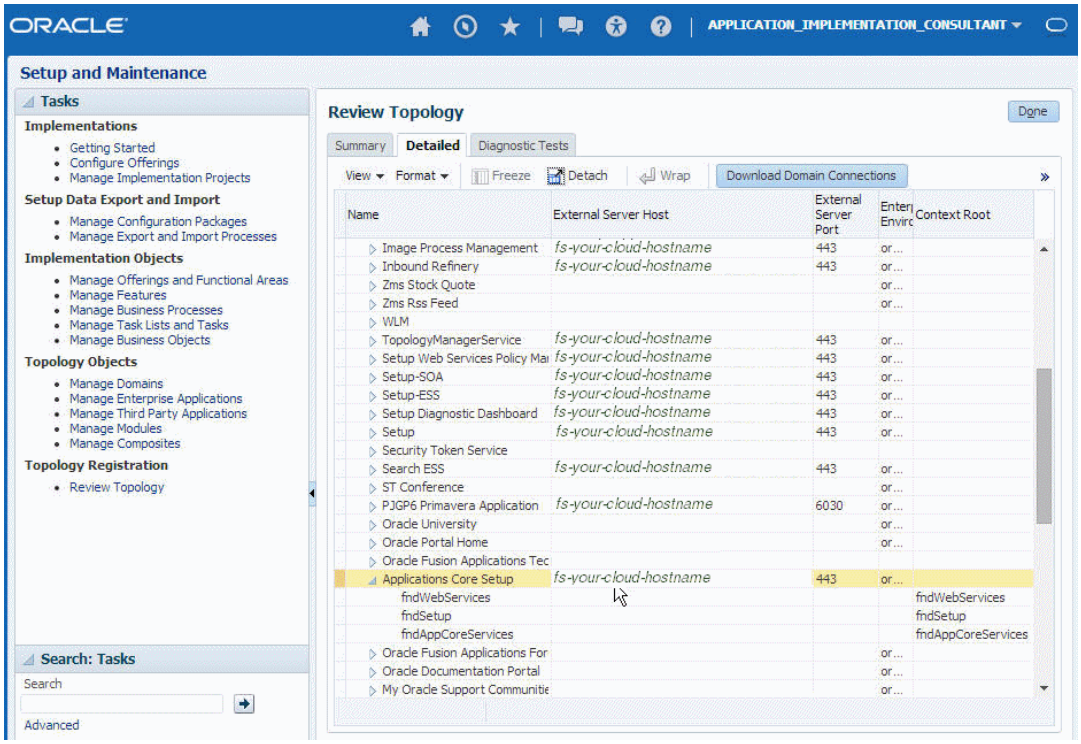
   The tab shows the list of domains configured in the cloud instance.



5. Map the token name for the service path value to the domain name in the Topology Manager:

Token Name in Service Path	Domain Name
atf_server	CommonDomain
crm_server	CRMDomain
fin_server	FinancialDomain
hcm_server	HCMDomain
ic_server	ICDomain
prc_server	ProcurementDomain
prj_server	ProjectsDomain
scm_server	SCMDomain

6. Expand the domain name and select any external virtual host and port for the J2EE applications that are deployed on the domain. In the sample window, the values for this particular instance are **fs-your-cloud-hostname** and **443**, respectively.



7. Replace the *domainName_server*:*PortNumber* with the external virtual host and port identified in the previous step. For example:

   ```
   https://fs-your-cloud-hostname:port/fndAppCoreServices/ServiceCatalogService?wsdl
   ```

Obtain the Service Catalog Service WSDL For Release 12

To obtain the physical endpoint of your instance, perform the following steps:

1. Log in to the Fusion Applications home page. For example:

   ```
   https://acme.fs.us2.oraclecloud.com/homePage/faces/FuseWelcome
   ```

   Where `acme` is the system name and `fs` is a Fusion Applications domain.

2. Copy `https://acme.fs.us2.oraclecloud.com/` and append `fndAppCoreServices/ServiceCatalogService?WSDL`. For example:

   ```
   https://acme.fs.us2.oraclecloud.com/fndAppCoreServices/ServiceCatalogService?WSDL
   ```

Obtain the Interface Catalog URL

The interface catalog URL takes the following format:

```
https://fusxxxx-fs-ext.us.oracle.com/helpPortalApi/otherResources/latest/interfaceCatalogs
```

For Fusion Applications Releases 13 and Later

Obtain the Oracle Fusion Applications Release 13 and later service catalog service WSDLs and interface catalog URLs through the following methods.

- Obtain the Service Catalog Service WSDL
- Obtain the Interface Catalog URL

Obtain the Service Catalog Service WSDL

To obtain the physical endpoint of your instance, perform the following steps:

1. Log in to the Fusion Applications home page. For example:

   ```
   https://acme.fa.us6.oraclecloud.com/fscmUI/faces/FuseWelcome
   ```

   Where `acme` is the system name and `us6` is the data center.
2. Copy `https://acme.fa.us6.oraclecloud.com/` and append it with `fscmService/ServiceCatalogService?WSDL`. For example:

   ```
   https://acme.fs.us2.oraclecloud.com/fscmService/ServiceCatalogService?WSDL
   ```

Obtain the Interface Catalog URL

The interface catalog URL takes the following format:

```
https://fusxxxx-fa-ext.us.oracle.com/fscmRestApi/otherResources/latest/
interfaceCatalogs
```

Create a Connection

Before you can build an integration, you must create the connections to the applications with which you want to share data.

> **Note**
>
> You can also create a connection in the integration canvas. See Define Inbound Triggers, Outbound Invokes, and Actions.

To create a connection in Oracle Integration:

1. Decide where to start:
   - Work in a project (see why working with projects is preferred).
     a. In the navigation pane, click **Projects**.
     b. Select the project name.
     c. Click **Integrations**.
     d. In the **Connections** section, click **Add** if no connections currently exist or **+** if connections already exist. The Create connection panel opens.

- Work outside a project.
  a. In the navigation pane, click **Design**, then **Connections**.
  b. Click **Create**. The Create connection panel opens.

2. Select the adapter to use for this connection. To find the adapter, scroll through the list, or enter a partial or full name in the **Search** field.
3. Enter the information that describes this connection.

Element	Description
Name	Enter a meaningful name to help others find your connection when they begin to create their own integrations.
Identifier	Automatically displays the name in capital letters that you entered in the **Name** field. If you modify the identifier name, don't include blank spaces (for example, `SALES OPPORTUNITY`).
Role	Select the role (direction) in which to use this connection. **Note**: *Only* the roles supported by the adapter you selected are displayed for selection. Some adapters support all role combinations (trigger, invoke, or trigger and invoke). Other adapters support fewer role combinations. When you select a role, only the connection properties and security policies appropriate to that role are displayed on the Connections page. If you select an adapter that supports both invoke and trigger, but select only one of those roles, you'll get an error when you try to drag the adapter into the section you didn't select. For example, assume you configure a connection for the Oracle Service Cloud (RightNow) Adapter as only an **invoke**. Dragging the adapter to a **trigger** section in the integration produces an error.
Keywords	Enter optional keywords (tags). You can search on the connection keywords on the Connections page.
Description	Enter an optional description of the connection.
Share with other projects	**Note**: This field only appears if you are creating a connection in a project. Select to make this connection publicly available in other projects. Connection sharing eliminates the need to create and maintain separate connections in different projects. When you configure an adapter connection in a different project, the **Use a shared connection** field is displayed at the top of the Connections page. If the connection you are configuring matches the same type and role as the publicly available connection, you can select that connection to reference (inherit) its resources. See Add and Share a Connection Across a Project.

4. Click **Create**.

   Your connection is created. You're now ready to configure the connection properties, security policies, and (for some connections) access type.

5. Follow the steps to configure a connection.
   The connection property and connection security values are specific to each adapter. Your connection may also require configuration with an access type such as a private endpoint or an agent group.

6. Test the connection.

Configure Connection Properties

Enter connection information so your application can process requests.

1. Go to the **Properties** section.

   The fields that are displayed are based on your version of Oracle Integration.

2. For new connections created with the initial release of the simplified connections page on 2/18/20, the **HCM Cloud Host** field is displayed. Enter the Oracle HCM Cloud host name. For example:

   `https://customer_chosen_domain_name.fa.DC.oraclecloud.com`

   > **Note**
   >
   > The Oracle HCM Cloud host name can easily be derived from the Oracle HCM Cloud login URL. For example: `https://customer_chosen_domain_name.fa.DC.oraclecloud.com/fscmUI/faces/FuseWelcome`

3. For existing connections created prior to the initial release of the simplified connections page on 2/18/20, the **HCM Services Catalog WSDL URL** and **Interface Catalog URL** fields are displayed. Specify the URLs to use in this integration.

   a. In the **HCM Services Catalog WSDL URL** field, specify the URL to use in this integration.

   b. In the **Interface Catalog URL** field, optionally specify the URL to consume Oracle HCM Cloud REST API business resources.

      If the interface catalog URL is not specified, the **Subscribe to Updates (via ATOM Feed)** option does not appear for selection on the Actions page of the Adapter Endpoint Configuration Wizard.

Configure Connection Security

Configure security for your Oracle HCM Cloud Adapter connection by selecting the security policy and security token.

1. Go to the **Security** section.

2. Select the security policy to use. Based on your selection, the page is refreshed to display various login credential fields.

Element	Description
Username Password Token With PGP Key Support	Specify the following details to upload an encrypted file to Oracle WebCenter Content (Universal Content Management (UCM). The supported algorithm for the public key is RSA for encryption and key size should be 1024 bits long. • **Username**: Enter the username. • **Password**: Enter the password. • **Client PGP Private Key**: Click  , then browse for and upload the private key to decipher encrypted content. The supported algorithm for the private key is RSA for decryption and the key size must be 1024 bits in size. • **Client PGP Passphrase**: Enter the passphrase registered with the PGP private key. • **PGP Public Key for UCM Upload**: Click  , then browse for and upload the public key to encrypt the file. The PGP public key must already be created. See Upload Files to Oracle WebCenter Content.
Username Password Token	You receive the username and password to enter when subscribing to Oracle HCM Cloud. • **Username**: Enter the username. • **Password**: Enter the password.

Element	Description

OAuth Authorization Code Credentials	• **Client ID**: Enter the client identifier (ID) issued during OAuth client application creation. The client ID identifies the client (the software requesting an access token) making the request. See Perform Prerequisites to Set Up the OAuth Authorization Code Credentials Security Policy.
	• **Client Secret**: Enter the client secret issued during OAuth client application creation. See Perform Prerequisites to Set Up the OAuth Authorization Code Credentials Security Policy.
	• **Authorization Code URI**: Enter the URI from which to request the authorization code. This endpoint is used to initiate the OAuth authentication and authorization process during which a user is directed to the OAuth server to provide credentials, to review granted permissions, and to provide consent.
	`https://Identity Domain_URL/oauth2/v1/authorize`
	• **Access Token URI**: Enter the URI to use for the access token. A request must be sent to this URI to obtain an access token.
	`https://Identity Domain_URL/oauth2/v1/token`
	• **Scope**: Enter the scopes specified during OAuth client application creation:
	– The URL that corresponds to the federated Oracle Fusion Application instance.
	– `offline_access`
	`https://FA_URL:443/offline_access`
	Scopes enable you to specify the type of access you need. Scopes limit access for the OAuth token. They do not grant any additional permission beyond that which the user already possesses. See Perform Prerequisites to Set Up the OAuth Authorization Code Credentials Security Policy.
	• **Client Authentication**: You can optionally configure OAuth flows with client authentication. This is similar to the Postman user interface feature for configuring client authentication.
	– **Send client credentials as basic auth header**: Pass the client ID and client

Element	Description
	secret in the header as basic authentication. – **Send client credentials in body**: Pass the client ID and client secret in the body as form fields. When configuration is complete, perform the following steps: a. Click **Provide Consent** to test the OAuth flow. b. If the identity domain Oracle Integration and Oracle Fusion Applications users are different, log in to the respective instance when prompted. **Note**: You are not prompted to log in if these users are the same. c. Return to the Connections page and click **Test**. **Note**: If you receive an `Unauthorized 401` error when testing your connection with a nonfederated user account, you may be logged in with the wrong user account.
OAuth using JWT User Assertion **Note**: • This policy is typically used on behalf of a user. • This policy supports the propagation of a user identity between systems. See Propagate OAuth User Identity Between Services.	• **Access token URI** — Enter the URL to which to send a request to obtain the access token. For example: `https://accounts.google.com/o/oauth2/token` • **JWT headers in JSON format** — Upload the JWT header file in JSON format. • **JWT payload in JSON format** — Upload the JWT payload file in JSON format. • **JWT private key alias** — Enter the JWT private key alias. This is the same alias you specified when uploading the signing key certificate on the Certificates page. • **Scope** — (Optional) Enter the scopes. • **Access token request** — (Optional) Enter the request to obtain the access token. The format you specify can vary by service provider.

Configure the Endpoint Access Type

Configure access to your endpoint. Depending on the capabilities of the adapter you are configuring, options may appear to configure access to the public internet, to a private endpoint, or to an on-premises service hosted behind a fire wall.

- Select the Endpoint Access Type
- Ensure Private Endpoint Configuration is Successful

Select the Endpoint Access Type

1. Go to the **Access type** section.

2. Select the option for accessing your endpoint.

Option	This Option Appears If Your Adapter Supports ...
Public gateway	Connections to endpoints using the public internet.
Private endpoint	Connections to endpoints using a private virtual cloud network (VCN). **Note**: To connect to private endpoints, you must complete prerequisite tasks in the Oracle Cloud Console. Failure to do so results in errors when testing the connection. See Connect to Private Resources in *Provisioning and Administering Oracle Integration 3* and Troubleshoot Private Endpoints in *Using Integrations in Oracle Integration 3*.

Ensure Private Endpoint Configuration is Successful

- To connect to private endpoints, you must complete prerequisite tasks in the Oracle Cloud Console. Failure to do so results in errors when testing the connection. See Connect to Private Resources in *Provisioning and Administering Oracle Integration 3*.
- When configuring an adapter on the Connections page to connect to endpoints using a private network, specify the fully-qualified domain name (FQDN) and *not* the IP address. If you enter an IP address, validation fails when you click **Test**.

Test the Connection

Test your connection to ensure that it's configured successfully.

1. In the page title bar, click **Test**. What happens next depends on whether your adapter connection uses a Web Services Description Language (WSDL) file. Only some adapter connections use WSDLs.

If Your Connection...	Then...
Doesn't use a WSDL	The test starts automatically and validates the inputs you provided for the connection.
Uses a WSDL	A dialog prompts you to select the type of connection testing to perform: • **Validate and Test**: Performs a full validation of the WSDL, including processing of the imported schemas and WSDLs. Complete validation can take several minutes depending on the number of imported schemas and WSDLs. No requests are sent to the operations exposed in the WSDL. • **Test**: Connects to the WSDL URL and performs a syntax check on the WSDL. No requests are sent to the operations exposed in the WSDL.

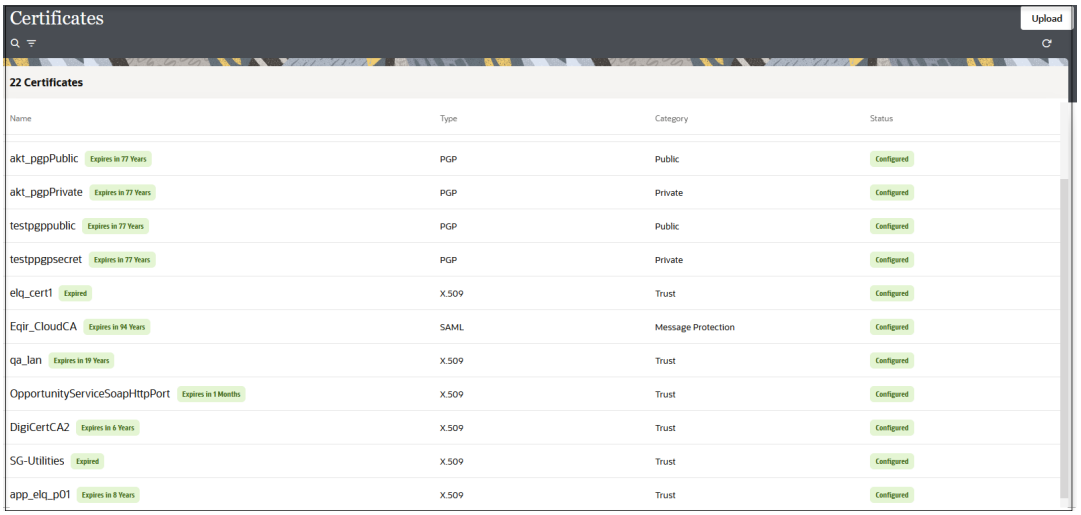
2. Wait for a message about the results of the connection test.
   - If the test was successful, then the connection is configured properly.
   - If the test failed, then edit the configuration details you entered. Check for typos and verify URLs and credentials. Continue to test until the connection is successful.
3. When complete, click **Save**.

Upload a Certificate to Connect with External Services

Certificates allow Oracle Integration to connect with external services. If the external service/endpoint needs a specific certificate, request the certificate and then import it into Oracle Integration.

If you make an SSL connection in which the root certificate does not exist in Oracle Integration, an exception error is thrown. In that case, you must upload the appropriate certificate. A certificate enables Oracle Integration to connect with external services. If the external endpoint requires a specific certificate, request the certificate and then upload it into Oracle Integration.

1. Sign in to Oracle Integration.
2. In the navigation pane, click **Settings**, then **Certificates**.
   All certificates currently uploaded to the trust store are displayed on the Certificates page.
3. Click **Filter** to filter by certificate expiration date, status, and type. Certificates installed by the system cannot be deleted.



4. Click **Upload** at the top of the page.
   The Upload certificate panel is displayed.
5. Enter an alias name and optional description.
6. In the **Type** field, select the certificate type. Each certificate type enables Oracle Integration to connect with external services.
   - Digital Signature
   - X.509 (SSL transport)
   - SAML (Authentication & Authorization)
   - PGP (Encryption & Decryption)
   - Signing key

Digital Signature

The digital signature security type is typically used with adapters created with the Rapid Adapter Builder. See Learn About the Rapid Adapter Builder in Oracle Integration in *Using the Rapid Adapter Builder with Oracle Integration 3*.

1. Click **Browse** to select the digital certificate. The certificate must be an X509Certificate. This certificate provides inbound RSA signature validation. See RSA Signature Validation in *Using the Rapid Adapter Builder with Oracle Integration 3*.
2. Click **Upload**.

X.509 (SSL transport)

1. Select a certificate category.
   a. **Trust**: Use this option to upload a trust certificate.
      i. Click **Browse**, then select the trust file (for example, `.cer` or `.crt`) to upload.
   b. **Identity**: Use this option to upload a certificate for two-way SSL communication.
      i. Click **Browse**, then select the keystore file (`.jks`) to upload.
      ii. Enter the comma-separated list of passwords corresponding to key aliases.

      > **Note**
      >
      > When an identity certificate file (`.jks`) contains more than one private key, all the private keys must have the same password. If the private keys are protected with different passwords, the private keys cannot be extracted from the keystore.

      iii. Enter the password of the keystore being imported.
   c. Click **Upload**.

SAML (Authentication & Authorization)

1. Note that **Message Protection** is automatically selected as the only available certificate category and cannot be deselected. Use this option to upload a keystore certificate with SAML token support. Create, read, update, and delete (CRUD) operations are supported with this type of certificate.
2. Click **Browse**, then select the certificate file (`.cer` or `.crt`) to upload.
3. Click **Upload**.

PGP (Encryption & Decryption)

1. Select a certificate category. Pretty Good Privacy (PGP) provides cryptographic privacy and authentication for communication. PGP is used for signing, encrypting, and decrypting files. You can select the private key to use for encryption or decryption when configuring the stage file action.
   a. **Private**: Uses a private key of the target location to decrypt the file.
      i. Click **Browse**, then select the PGP file to upload.
      ii. Enter the PGP private key password.
   b. **Public**: Uses a public key of the target location to encrypt the file.
      i. Click **Browse**, then select the PGP file to upload.
      ii. In the **ASCII-Armor Encryption Format** field, select **Yes** or **No**.
         - **Yes** shows the format of the encrypted message in ASCII armor. ASCII armor is a binary-to-textual encoding converter. ASCII armor formats encrypted

messaging in ASCII. This enables messages to be sent in a standard messaging format. This selection impacts the visibility of message content.

- **No** causes the message to be sent in binary format.

iii. From the **Cipher Algorithm** list, select the algorithm to use. Symmetric-key algorithms for cryptography use the same cryptographic keys for both encryption of plain text and decryption of cipher text. The following supported cipher algorithms are FIPS-compliant:

- AES128
- AES192
- AES256
- TDES

c. Click **Upload**.

Signing key

A signing key is a secret key used to establish trust between applications. Signing keys are used to sign ID tokens, access tokens, SAML assertions, and more. Using a private signing key, the token is digitally signed and the server verifies the authenticity of the token by using a public signing key. You must upload a signing key to use the OAuth Client Credentials using JWT Client Assertion and OAuth using JWT User Assertion security policies in REST Adapter invoke connections. Only PKCS1- and PKCS8-formatted files are supported.

1. Select **Public** or **Private**.
2. Click **Browse** to upload a key file.
   If you selected **Private**, and the private key is encrypted, a field for entering the private signing key password is displayed after key upload is complete.
3. Enter the private signing key password. If the private signing key is not encrypted, you are not required to enter a password.
4. Click **Upload**.

Refresh Integration Metadata

You can manually refresh the currently-cached metadata available to adapters that have implemented metadata caching.

Metadata changes typically relate to customizations of integrations, such as adding custom objects and attributes to integrations. There may also be cases in which integrations have been patched, which results in additional custom objects and attributes being added. This option is similar to clearing the cache in your browser. Without a manual refresh, a staleness check is only performed when you drag a connection into an integration. This is typically sufficient, but in some cases you may know that a refresh is required. For these cases, the **Refresh Metadata** menu option is provided.

> **Note**
>
> The **Refresh Metadata** menu option is only available with adapters that have implemented metadata caching.

1. Decide where to start:

- Work in a project (see why working with projects is preferred).
  a. In the navigation pane, click **Projects**.
  b. Select the project name.
  c. Click **Integrations**.
  d. In the **Connections** section, hover over the adapter connection to refresh.
- Work outside a project.
  a. In the navigation pane, click **Design**, then **Connections**.
  b. Hover over the adapter connection to refresh.

2. Click **Actions**, then select **Refresh metadata**.

   If successful, the following message is displayed.

   ```
   Metadata refresh for connection connection_name has been initiated
   successfully.
   ```

3

Add the Oracle HCM Cloud Adapter Connection to an Integration

When you drag the Oracle HCM Cloud Adapter into the trigger or invoke area of an integration, the Adapter Endpoint Configuration Wizard appears. This wizard guides you through the configuration of the Oracle HCM Cloud Adapter endpoint properties.

These topics describe the wizard pages that guide you through configuration of the Oracle HCM Cloud Adapter as a trigger or invoke in an integration.

Topics:

Basic Info Page

You can enter a name and description on the Basic Info page of each adapter in your integration.

Element	Description
What do you want to call your endpoint?	Provide a meaningful name so that others can understand the responsibilities of this connection. You can include English alphabetic characters, numbers, underscores, and hyphens in the name. You can't include the following characters: • No blank spaces (for example, `My Inbound Connection`) • No special characters (for example, `#;83&` or `righ(t)now4`) except underscores and hyphens • No multibyte characters
What does this endpoint do?	Enter an optional description of the connection's responsibilities. For example: `This connection receives an inbound request to synchronize account information with the cloud application.`

Trigger Request Page

Enter the Oracle HCM Cloud trigger request values for your integration. The values you specify start the integration.

Element	Description
Select a Business Object	Select the business object from the Oracle HCM Cloud application to receive as a request that starts the integration.
Filter by object name	Type the initial letters of the name to filter the display of business objects.

Trigger Response Page

Select the Oracle HCM Cloud trigger response business object for your integration.

Element	Description
Filter by object name	Type the initial letters to filter the display of business objects.
Select a Business Object	Select the business object to receive from the Oracle HCM Cloud application as a response.

Invoke Action Page

Select the Oracle HCM Cloud invoke action for your integration.

Element	Description
What Would You Like to do with Oracle HCM Cloud Adapter	• **Query, Create, Update, or Delete Information**: Select to query business objects such as employee records or perform operations for employee onboarding, data synchronization, and so on. • **Subscribe to Updates (via Atom Feed)**: Select to receive the latest Atom feed updates since a specific date for new hires, new jobs, and so on. **Note** The **Subscribe to Updates (via Atom Feed)** option is currently supported for Releases 11 and higher of Oracle HCM Cloud. • **Import Bulk Data using HCM Data Loader (HDL)**: Select to import bulk data using the HCM Data Loader. • **Receive Files from HCM Cloud**: Select to receive several records as a data file on payroll records, time sheets, and others. • **Send Files to HCM Cloud**: Select to upload files to Oracle WebCenter Content (Universal Content Manager) in encrypted or unencrypted format.

Invoke Operation Page

Enter the Oracle HCM Cloud invoke operation values for your integration.

See the appropriate section based on your selection on the Actions page:

- Business Objects, Services, and Business (REST) Resources
- Atom Feeds
- Data Extracts
- File Upload to WebCenter (UCM)
- HCM Data Loader

Business Objects, Services, and Business (REST) Resources

If you select **Query, Create, Update, or Delete Information** on the Actions page, the following fields are displayed.

Element	Description
Browse by	Select to browse by business object or service. There is a one-to-one correspondence between the business object and service. The service acts on the business document. • **Business Objects**: Select to browse a list of available business objects. • **Services**: Select to browse a list of available services. • **Business (REST) Resources**: Select to browse a list of available business (REST) resources exposed by the interface catalog URL. Select the business object or service.
Select a Business Object (displayed if **Business Objects** is selected)	Select the business object to use.
Select a Service (displayed if **Services** is selected)	Select the service to use.
Select a Service Application (displayed if **Business (REST) Resources** is selected)	Select the business (REST) resource to use. You can then click **Browse and configure a child response** to select the corresponding child business resources of that parent to use. **Note**: Existing integrations created prior to the introduction of this feature can be edited to select parent business resources and their corresponding child business resources. See About the REST APIs in *REST API for Oracle Fusion Cloud HCM* for details about supported REST resources.
Browse and configure a child response	Click to access a page to select the following: • The child and subchild business resources of the selected parent business resource • The operation to perform on the child and subchild business resources After you click **Ok**, the link name changes to **View and edit the configuration of a child resource**. Both the parent and child business resources are displayed on the Summary page. To reset to your original selections, click this link, then click **Reset**.
Filter by *type*	Type the initial letters to filter the display of business objects, services, or business (REST) resources.
Select the Operation to Perform on the Business Object/Resource or Service	Select the operation to perform on the selected business object, business (REST) resource, or service. **Note**: If you select **get**, only the following query parameters are supported: • **expand** • **fields** • **onlyData**

Element	Description
Life Cycle	Displays the state of the selected business object or service. Deprecated indicates the business document is nearing the end of use and must be used with caution.
Name	Displays the name of the selected business object or service.
Description	Displays the description of the selected business object or service.

Atom Feeds

If you selected the **Subscribe to Updates (via ATOM Feed)** option on the Actions page, the following fields are displayed.

> **Note**
>
> Any new Atom feed endpoint that uses Employee-related feeds (for example, new hire, assignment, termination, and updates) with a business object now uses the Workers resource in both design time and runtime. The Workers resource is visible in the mapper.
> Any existing Atom feed endpoint continues to use the Emps resource for employee-related feeds with a business object for backward compatibility and to avoid the need for remapping.

Element	Description
Select an Atom Feed	Select the feed of interest to the downstream application.
Learn more about HCM Cloud ATOM feeds	Click to access a page that describes how Atom feed subscriptions work.
Max entries to process	Select the number of feeds to process.
Process Future Dated Entries Immediately	Select to process future-dated feeds immediately. If not selected, future-dated feeds are processed when the effective date is reached. The integration is responsible for handling not-yet-effective, future-dated entries. The elements that appear in the mapper are different based on your selection. If **Process Future Dated Updates Immediately** is not selected, the request mapper shows the **updated-min** element under **ApplicationPullParameter**. If **Process Future Dated Updates Immediately** is selected, the request mapper shows the **published-min** element under **ApplicationPullParameter**. This makes future-dated Atom entries appear in their respective Atom feed immediately once the future-dated action is taken.

Element	Description
Include Business Object in ATOM Feeds	Select this checkbox to send an HTTP request for each entry in the feed to the ATOM server to fetch the business object snapshot. • If not selected, changed and context attributes are used during design time and runtime, the operation name in the mapper does not have the suffix **WithBO**, and the business object is not shown under the operation name element. • If selected, context attributes, changed attributes, and business object snapshots are used. The maximum number of entries to process is set to 1000 entries and cannot be changed. This sets the value of the query parameter `page-size` that is sent as part of the request to the Atom server to get the feed. ATOM server returns a maximum of 1000 entries as part of the feed.

Data Extracts

If you selected the **Receive Files from HCM Cloud** option on the Actions page, the following questions are displayed to complete the configuration.

Element	Description
What is the Integration Name for HCM Extracts	Specify the name. The HCM extract with a matching integration name is downloaded to Oracle Integration.
Specify release date-time of extract	Specify the time of the extract. The extracts released after the specified date-time are eligible for download. A sample specified date-time value is `6/22/17 5:04 AM`.
Select actions that need to perform on extract in following list	Select the actions to perform on the extract: • **Decrypt the extract**: Extract is decrypted if it is in encrypted form and PGP information is present in the connection configuration. • **Unzip the extract**: Extract is unzipped if it is in a zipped format. The selected actions are performed on the extract after download to Oracle Integration.

File Upload to WebCenter (UCM)

If you selected **Send Files to HCM Cloud** on the Actions page, select the security group and doc account required for uploading the file.

Element	Description
Security Group	Select the security group in which to upload the file. A security group is a set of files grouped under a unique name. Every file in the content server repository belongs to a security group. Access to security groups is controlled by permissions assigned to roles on the content server. Roles are assigned to users where they are maintained in Oracle Fusion Applications. The default security group in Fusion Applications is FAFusionImportExport.
Doc Account	Select the doc account to assign to the file. In Fusion Applications, every content item has an account assigned to it. You must have the appropriate permission to the account such as read and/or write. The access to the document is the intersection between account permissions and security group permissions. There are several Fusion Applications accounts.
Encrypt the File	Select this checkbox to encrypt the file before upload to UCM. To select this checkbox, you must have selected to encrypt the file when configuring the Oracle HCM Cloud Adapter connection on the Connections page. See Configure Connection Security. **Note**: The Oracle HCM Cloud Adapter does not support file encryption during an HDL import operation. The **HCM Data Loader (HDL) File Encryption** option only denotes whether or not the file being uploaded through the HDL operation is PGP-encrypted. If there is a requirement to upload an encrypted file, the encryption must be done using the stage file action **Encrypt file** operation before the HCM HDL endpoint is called. See Configure a Stage File Action in *Using Integrations in Oracle Integration 3*.

HCM Data Loader

If you selected **Import Bulk Data using HCM Data Loader (HDL)** on the Actions page, select the operation to perform against the target HCM Data Loader application.

Element	Description
Submit an HCM Data Loader Job	Select to trigger the upload of the data (`.dat`) file from Oracle Integration into Oracle WebCenter Content. The Oracle HCM Cloud Adapter uploads the ZIP file containing the `.dat` file to Oracle WebCenter Content and invokes the HCM Data Loader importAndLoad operation, which returns an HCM Data Loader process ID. The `.dat` file must be HDL-compliant. If you select this operation, the page refreshes to display the following fields: • **Security Group**: Select the security group. • **Doc Account**: Select the documentation account. • **HCM Data Loader (HDL) File Encryption**: Select **None** or **PGP Unsigned**, which provides Pretty Good Privacy (PGP) unsigned encryption of the `.dat` file.
Query the Status of an HCM Data Loader Job	Select to retrieve the status of an HCM Data Loader job submitted by this integration. The Oracle HCM Cloud Adapter invokes the HCM Data Loader getDataSetStaus operation to get the status of the HCM Data Loader process.

A use case that describes importing bulk data using the HCM Data Loader (HDL) is provided. See Import Bulk Data with the HCM Data Loader (HDL).

Invoke Child Resources Page

Select the child resources to include with the parent resource selected on the Operations page. This helps to minimize the size of the integration WSDL file. If you do not select any child resources, all child resources (including custom resources) associated with the parent resource are included by default in the integration WSDL file. This increases the size of the WSDL file and can cause memory issues in Oracle Integration. This page is only displayed if you select a top-level parent resource on the Operations page.

Select a maximum of ten child resources to include in either the request payload sent to the external API or the response message received from the external API. Do not select child resources that are not required for use by this integration.

Element	Description
Select Child Resources	Select the child resources to use. Only the child resources associated with the parent resource you selected on the Operations page are displayed for selection.
Your Selected Child Resources	Displays the selected child resources.

Invoke Descriptive and Extensible Page

Select the descriptive flexfield (DFF) or extensible flexfield (EFF) and associated contexts. A flexfield is a flexible data field that your organization can customize to meet your business needs without programming. It provides a set of placeholder fields (segments) associated with a business object.

Two types of flexfields are supported for selection:

- Descriptive flexfield: A field you customize to enter additional information for which your Oracle Fusion Applications product has not already provided a field.
- Extensible flexfield: Similar to a descriptive flexfield in that it provides a customizable expansion space that implementers (such as Oracle Fusion Applications users) can use to configure additional attributes (segments) without additional programming.

Element	Description
Select Flexfields(s)	Select a flexfield to see the configured contexts.
Select Context(s)	Select a maximum of 20 contexts to include in either the request payload sent to the external API or the response message received from the external API.
Number of Contexts Selected	Displays the number of selected contexts.

Summary Page

You can review the specified adapter configuration values on the Summary page.

Element	Description
Summary	Displays a summary of the configuration values you defined on previous pages of the wizard. The information that is displayed can vary by adapter. For some adapters, the selected business objects and operation name are displayed. For adapters for which a generated XSD file is provided, click the XSD link to view a read-only version of the file. To return to a previous page to update any values, click the appropriate tab in the left panel or click **Go back**. To cancel your configuration details, click **Cancel**.

4

Implement Common Patterns Using the Oracle HCM Cloud Adapter

You can use the Oracle HCM Cloud Adapter to implement the following common patterns.

Topics:

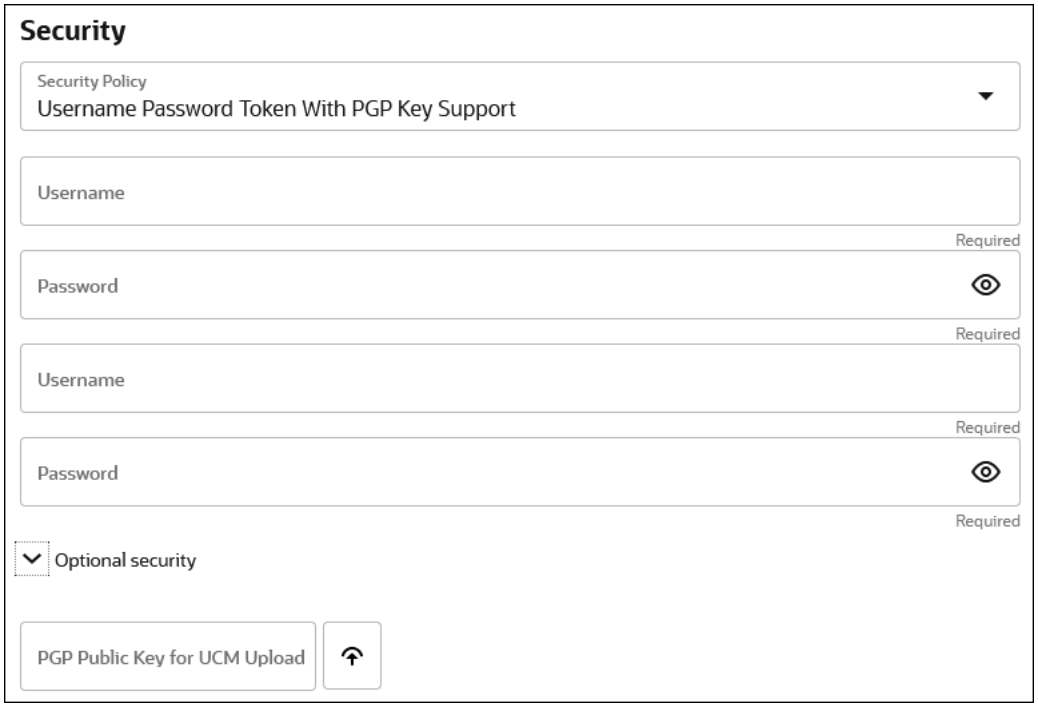
> **Note**
>
> Oracle Integration offers a number of prebuilt integrations, known as *recipes*, that provide you with a head start in building your integrations. You can start with a recipe, and then customize it to fit your needs and requirements. Depending upon the solution provided, a variety of adapters are configured in the prebuilt integrations.
> See the Recipes and Accelerators page on the Oracle Help Center.

Upload a File to Oracle WebCenter Content

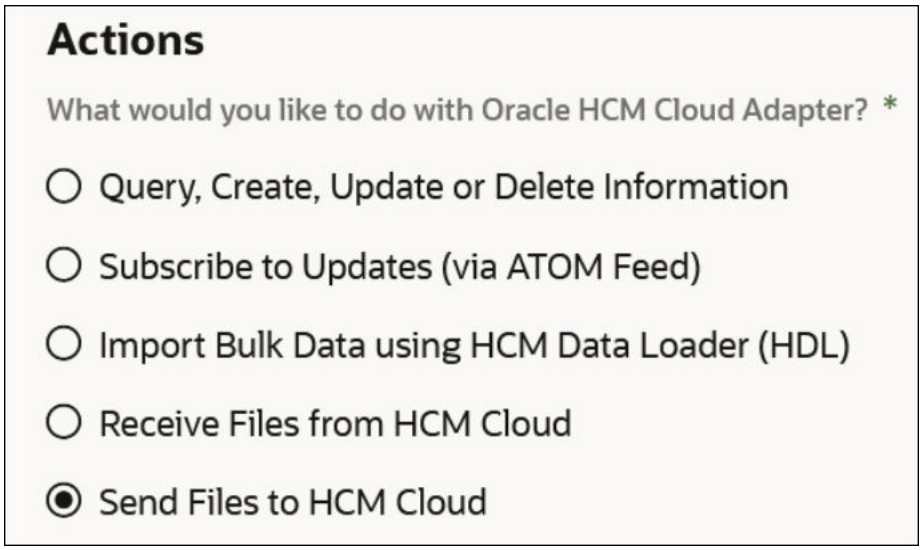
You can upload a file to Oracle WebCenter Content (Universal Content Manager) with the Oracle HCM Cloud Adapter. The file to upload can be either encrypted or unencrypted. This section provides a high-level overview for performing this scenario.

To upload a file to Oracle WebCenter Content:

1. Create an Oracle HCM Cloud Adapter connection with the **Invoke** role. During connection configuration, you can select to optionally encrypt the file to upload by selecting the **PGP Public Key for UCM Upload** checkbox.

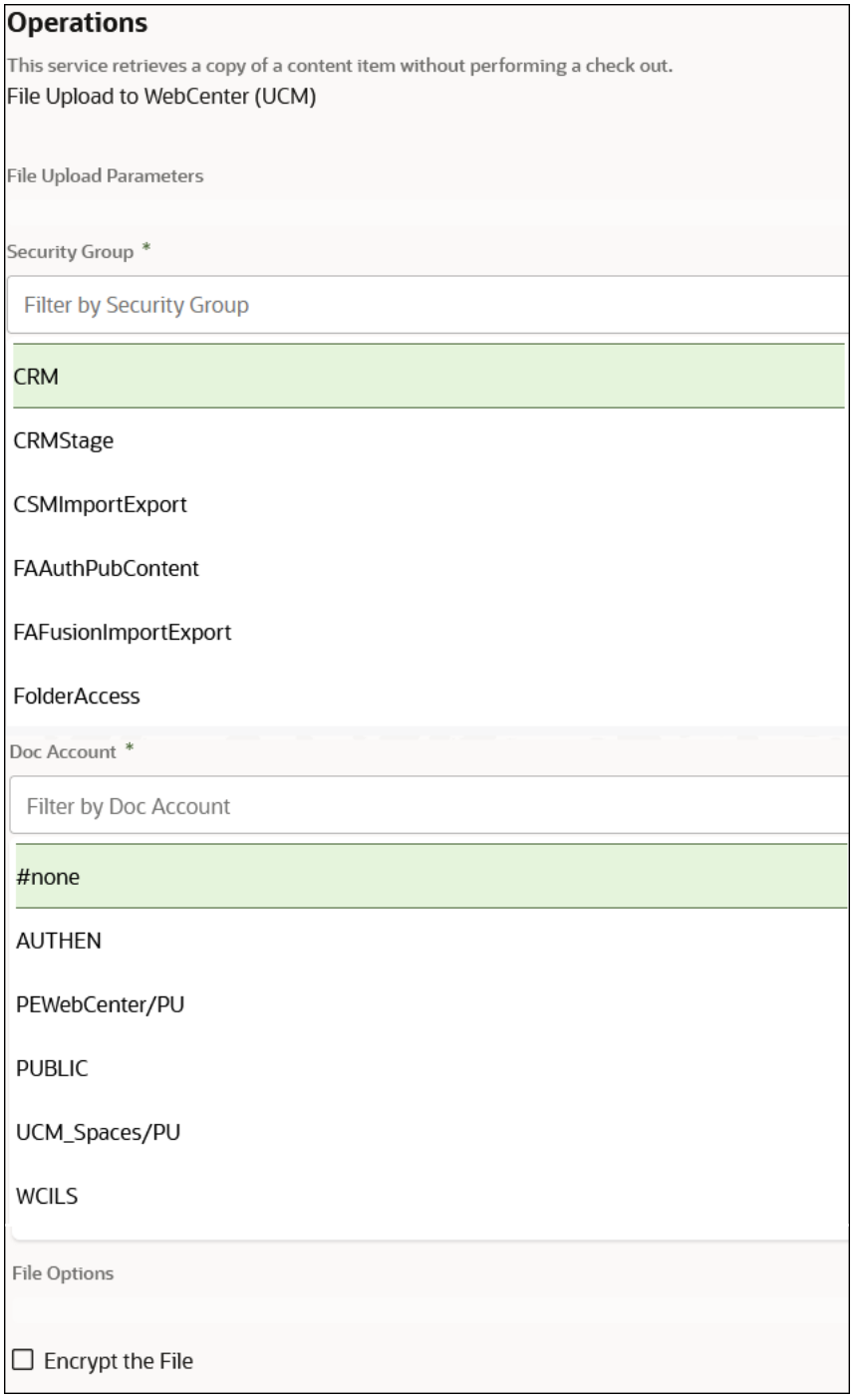


2. Create an orchestrated integration.
3. Drag the Oracle HCM Cloud Adapter to the invoke part of the integration canvas. This invokes the Adapter Endpoint Configuration Wizard.
4. On the Actions page, select **Send Files to HCM Cloud**.

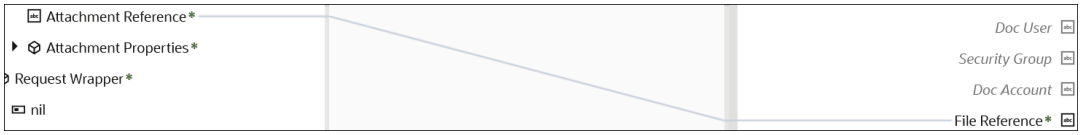


5. On the Operations page, select the following mandatory values:
   - **Security Group**: Select the security group to which the file to upload belongs. A security group is a set of files grouped under a unique name. Every file in the UCM server repository belongs to a security group. Access to security groups is controlled by permissions assigned to roles on the content server. Roles are assigned to users where they are maintained on Oracle Fusion Applications. The default security group in Fusion Applications **FAFusionImportExport**.

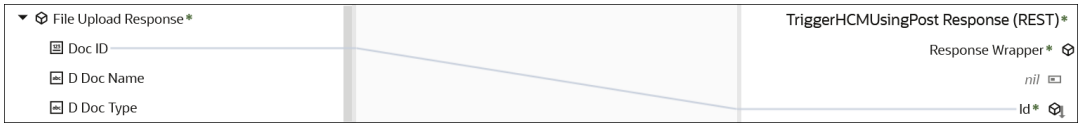
- **Doc Account**: Select the doc account for the file to upload. In Fusion Applications, every content item has an account assigned to it. You must have the appropriate permission to the account such as read and/or write. Access to the document is the intersection between account permissions and security group permissions.



6. If you selected to encrypt the file to upload on the Connections page in Step 1, select **Encrypt the File**.
7. Once the UCM file upload endpoint is saved, provide the reference in the mapper to the file to upload to UCM.



You can also override the security group and doc account that you previously set by hard coding new values in the mapper that receive reference during runtime.



8. Complete design of the integration.

Subscribe to Atom Feeds in a Scheduled Integration

This use case provides an overview of creating a scheduled orchestrated integration using the Oracle HCM Cloud Adapter to subscribe to Atom feeds. Atom feeds enable you to track changes made to feed-enabled resources in Oracle Global Human Resources Cloud. For any updates of interest to downstream applications such as new hires, terminations, employee transfers, and promotions, Oracle Global Human Resources Cloud publishes Atom feeds.

For this use case, the Oracle HCM Cloud Adapter is configured with the Atom feed **Employee Update**. This feed consists of three updates (**PrimaryPhoneNumber**, **CitizenshipStatus**, and **CitizenshipId**). An FTP Adapter is also configured to write any feed updates to an FTP server.

There are two types of entries published by Oracle Global Human Resources Cloud in any feed:

- Entries that are already effective.
- Entries that are effective in the future (known as future dated entries).

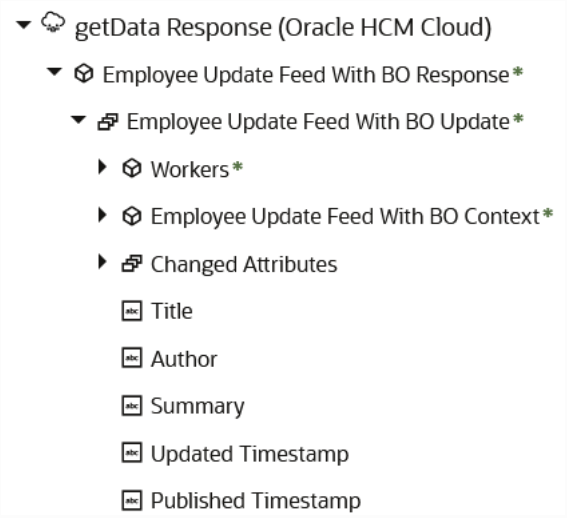
Design the integration based on these requirements. If the integration must process future dated entries, there are the following options:

- Future dated entries are processed on their effective dates.
- Future dated entries are processed immediately.

Note

Any new Atom feed endpoint that uses Employee-related feeds (for example, new hire, assignment, termination, and updates) with a business object now uses the Workers resource in both design time and runtime. The Workers resource is visible in the mapper.
Any existing Atom feed endpoint continues to use the Emps resource for employee-related feeds when it is already deployed with a business object for backward compatibility and to avoid the need for remapping. All new deployments with the Atom feed must use the Workers business object.

The following example shows the Employee update feed with a business object in the mapper:



Process Future Dated Entries on Their Effective Dates

Oracle Global Human Resources Cloud processes future dated entries on their effective dates. This use case provides an overview of how to design this type of integration.

1. On the Connections page, create and configure an Oracle HCM Cloud Adapter with the following details:
   a. Specify a name.
   b. In the Configure Connectivity dialog, specify *both* a service catalog WSDL URL and an interface catalog URL. If the interface catalog URL is not specified, the **Subscribe to Updates (via Atom Feed)** option is not displayed on the Actions page of the Adapter Endpoint Configuration Wizard.
   c. In the Credentials dialog, select the **Username Password Token** security policy and specify the login credentials.
2. On the Integrations page, create a scheduled orchestrated integration.
3. Drag the Oracle HCM Cloud Adapter into the integration canvas as an invoke connection and configure the Adapter Endpoint Configuration Wizard with the following details:
   a. On the Basic Info page, specify a name (for this example, `getData`).

b. On the Actions page, select **Subscribe to Updates (via Atom Feed)**.

c. On the Operations page, select the following:

- From the **Atom feed** list, select **Employee Update**.
- Select the **Include Business Object in Atom Feeds** checkbox to send an HTTP request for each entry in the feed to the Atom server to fetch the latest snapshot of the business resource. The checkbox behavior is as follows:
  - Not selected: Context and changed attributes are used during design time and runtime. The operation name in the mapper does not have the suffix **WithBO** and the business object is not shown under the operation name element.
  - Selected: Both context attributes and business object attributes are used during design time and runtime. The operation name in the mapper has the suffix **WithBO**.
- Select the maximum number of entries to process from the **Max entries to process** list.
  This sets the value of the **page-size** query parameter that is sent as part of the request to the Atom server to get the feed. The Atom feed size is limited by this number. The recommendation is to use a small number for this option and have the integration execute more frequently.

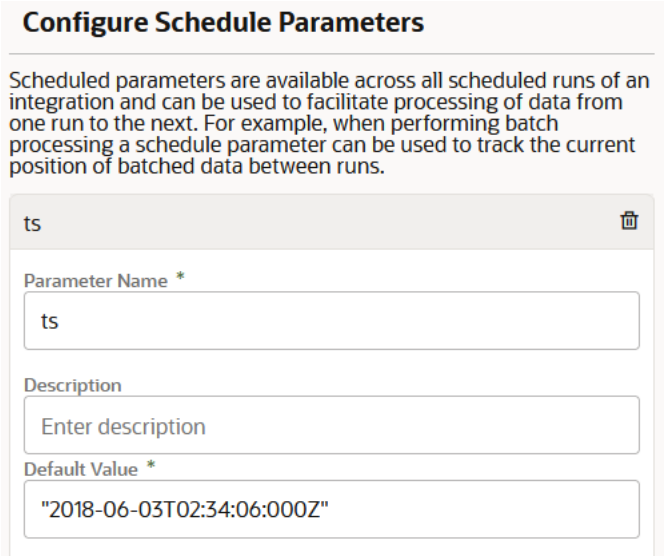
4. Double-click the schedule icon, and select **Edit** to create a schedule parameter (for this example, named **ts**). The initial value must be a timestamp to which to start. Entries are received during runtime from this timestamp onwards. The timestamp must be of following format:

```
YYYY-MM-DDTHH:MM:SS:sssZ e.g. 2018-06-03T02:34:06.000Z
```

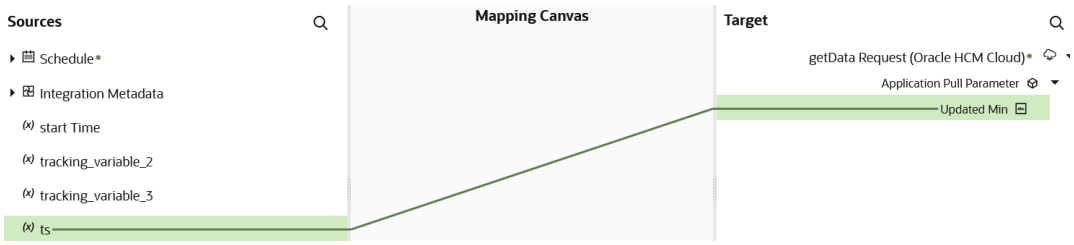
where:

- `YYYY`: Four-digit year
- `MM`: Two-digit month (`01`=January, and so on)
- `DD`: Two-digit day of the month (`01` through `31`)
- `hh`: Two digits of the hour (`00` through `23`) (am/pm NOT allowed)
- `mm`: Two digits of the minute (`00` through `59`)
- `ss`: Two digits of the second (`00` through `59`)
- `s`: Three digits representing a decimal fraction of a second (that is, milliseconds)

This parameter ensures that new entries are processed every time the integration is invoked. The timestamp of the last processed entry must be stored in this parameter. This value is available across invocations of the integration. For the first request, the **updated-min** query parameter value is blank, which returns the latest *n* number of entries in the feed, where *n* is the value selected for the **Max entries to process** option on the Operations page.

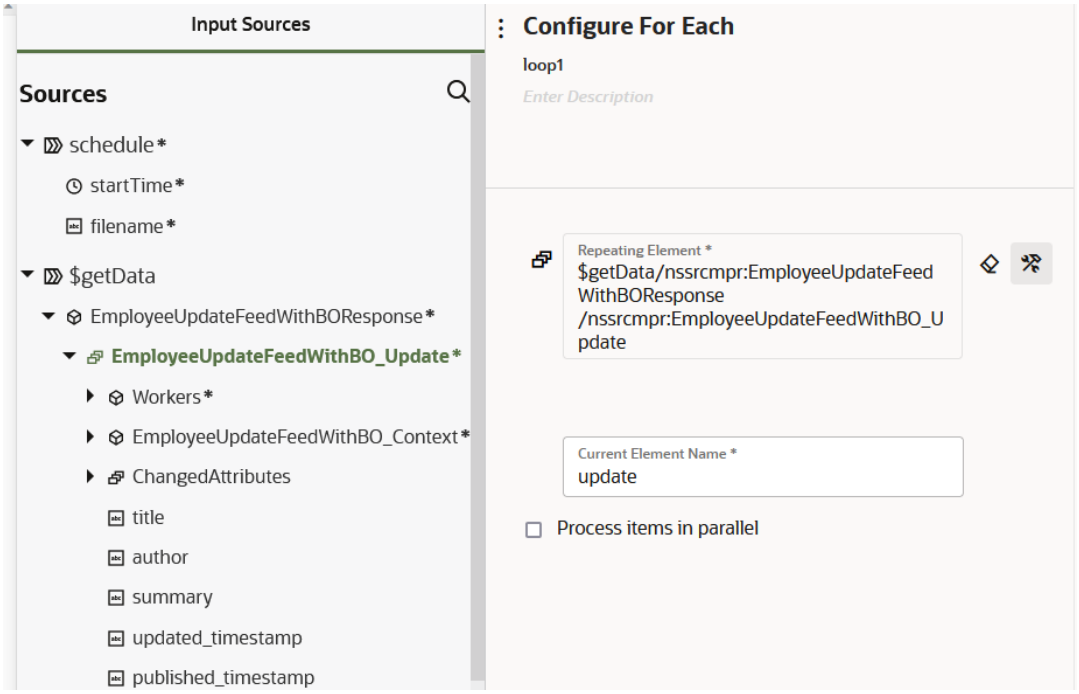


5. In the request mapper between the schedule and the Oracle HCM Cloud Adapter, map **$ts** (timestamp value) to **updated-min**. The Oracle HCM Cloud Adapter sends the **updated-min** query parameter to the Atom server when requesting a feed. The Atom server returns a feed with updated entries occurring after newer updates to the timestamp value. This enables new updates to be processed every time and prevents the same update from being processed multiple times.

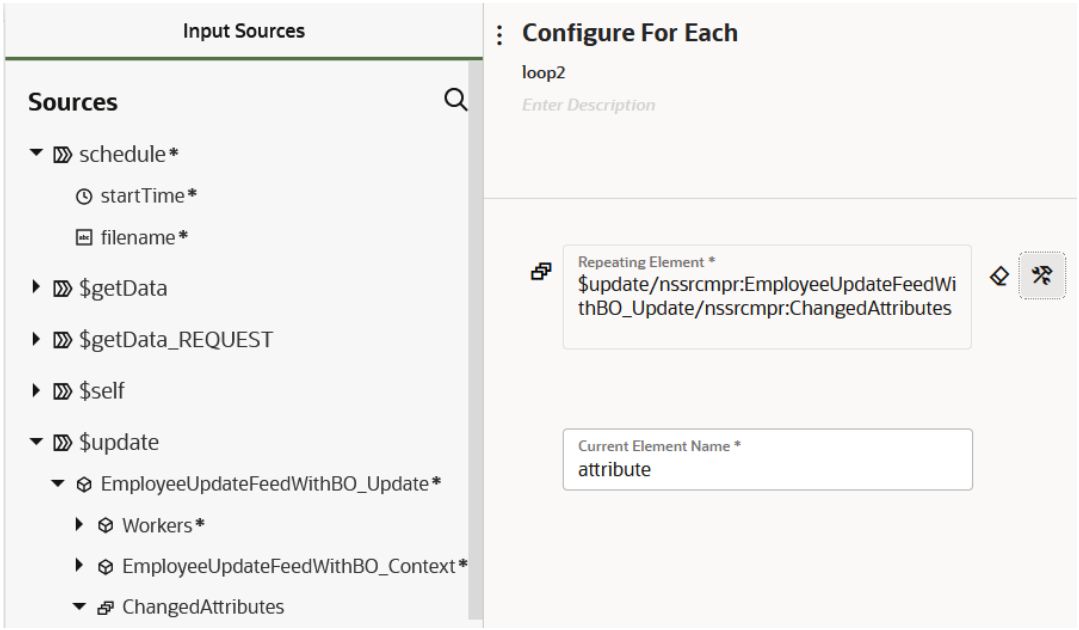


6. Add and configure a **For Each** action below the Oracle HCM Cloud Adapter in the integration canvas.

   Because the **Include Business Object in Atom Feeds** option was selected on the Operations Page, the mapper shows the business object (**Workers**), the context (**EmployeeUpdateFeedWithBO_Context**), the changed attributes (**ChangedAttributes**), and the timestamp. The timestamp is the updated element from the feed's entry. This is used to track processed entries. This is done with the help of the schedule parameter. The first **For Each** action iterates through the entries.

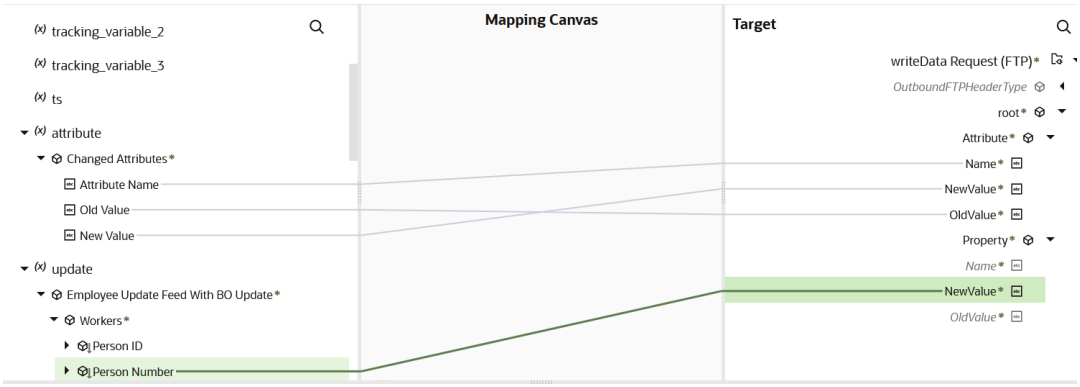


7. Add and configure a second **For Each** action inside the first **For Each** action to iterate through the changed attributes.



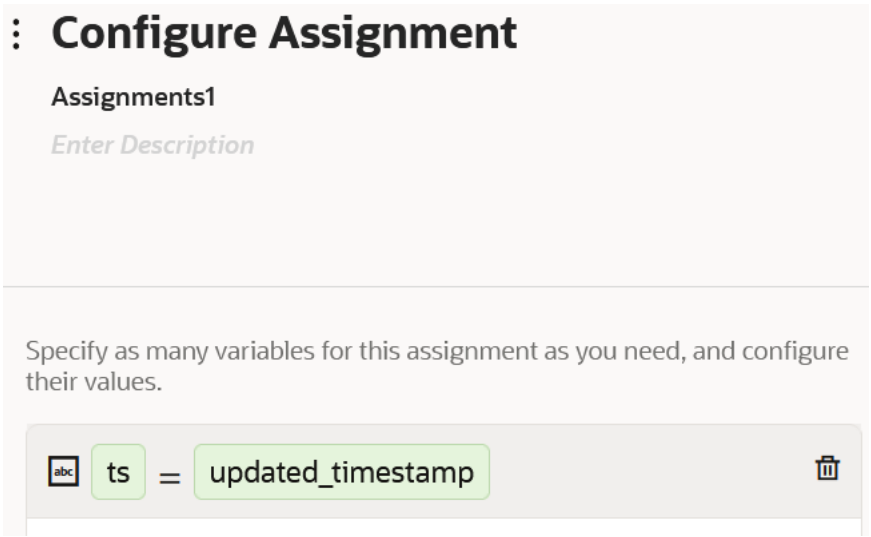
8. Add and configure an FTP Adapter inside the second **For Each** action in the integration.

   For each changed attribute, a file is written to the FTP server. You can also use a REST Adapter connection to send multiple patch requests to an endpoint with these changes.



At the end of the first loop (for this example, named **loop1**), the **ts** schedule parameter is updated with the timestamp of the entry. This is stored in the database and available for subsequent invokes. This value is persisted across subsequent integration runs and also across deactivations and activations. This is how the processed entries are tracked. Because the **ts** schedule parameter is used in the request mapping, in subsequent invokes it uses its value from the database. Therefore, only new entries are processed.

9. Add an **Assign** action at the bottom of the first **For Each** action (**loop1**). The assign activity is placed at the end of **loop1** for fault tolerance. If a fault occurs when writing to the FTP location, the updated timestamp of the last processed entry is used for the next invoke.



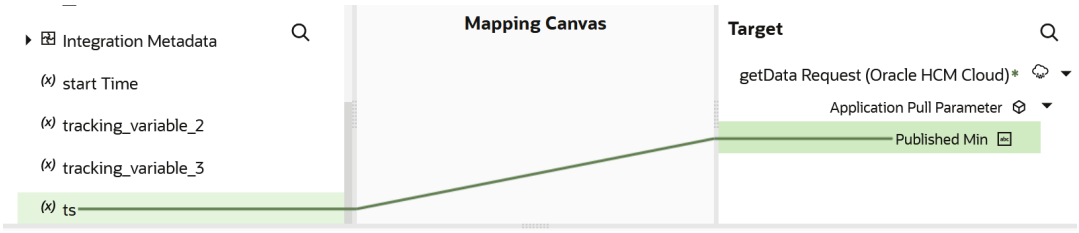
The **updated_timestamp** element is selected from the update element (entry in the feed) as the value for schedule parameter **ts**.

When the integration is invoked by the schedule, a file is created in the FTP directory for each of the changed attributes. The complete design of the integration is as follows:

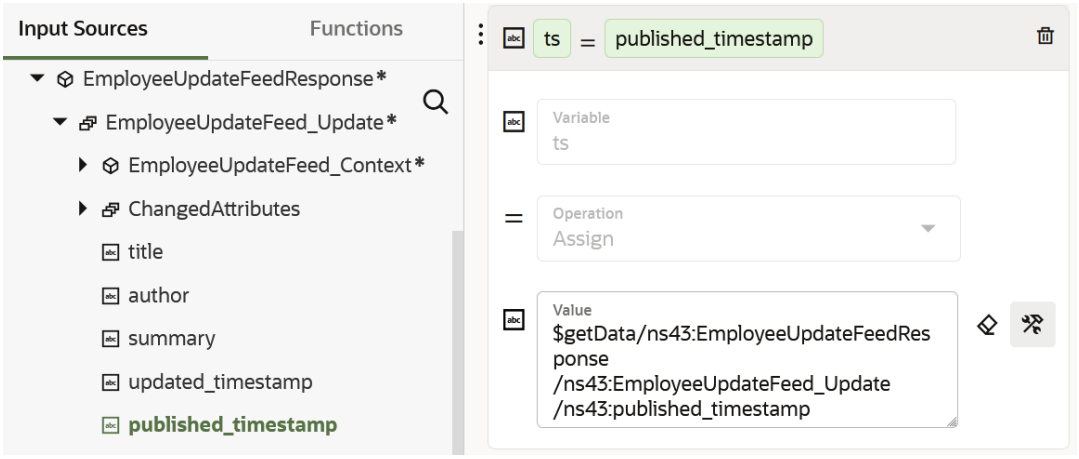
Process Future Dated Entries Immediately

Oracle Global Human Resources Cloud processes future dated entries immediately, as soon as they are published. This use case provides an overview of the differences between designing an integration to process future dated entries on their effective dates and designing an integration to process future dated entries immediately.

- On the Operations page, select **Process Future Dated Entries Immediately**. Enabling this checkbox changes the request map to have a different query parameter, **published-min**.
- In the request mapper between the schedule and the Oracle HCM Cloud Adapter, map **$ts** (timestamp value) to **published-min**.



- To make future dated entries available immediately in the feed, you must send **published_timestamp** as **published-min** in the request. **published_timestamp** must also be persisted in the **ts** scheduled parameter to process new entries in subsequent invokes of the flow. **ts** must store the **published_timestamp**.



Summary

In summary, the following changes are required in the integration for handling future dated entries.

Future Dated Entries Options	Request Parameters	Timestamp Stored in Schedule Parameter
Processed immediately	**published-min**	**published_timestamp**
Processed on their effective dates	**updated-min**	**updated_timestamp**

Configure the Extract Bulk Data Option in an Integration

You can use the extract bulk data option in an orchestrated integration. This section provides a high-level design of an integration using this feature.

> **Note**
>
> You must schedule and create the HCM data extract in Oracle HCM Cloud. It must be configured with a delivery option of type **WebCenter Content**. When configuring this delivery option, the **Integration Name** field must be populated with a unique value that is later specified in the **What is the Integration Name for HCM Extracts** field of the Operations page when configuring the Oracle HCM Cloud Adapter.

To configure an extract bulk data integration:

1. Create a scheduled, orchestrated integration pattern.
2. Add a schedule parameter to store the processed Document ID and initialize it with a value of `0`. An HCM extract is associated with a Document ID. Once the extract is processed by the integration, the ID is stored in the schedule parameter. This value is required as input by the data extract operation in the Oracle HCM Cloud Adapter.
3. Drag the Oracle HCM Cloud Adapter to the invoke section of the integration canvas and configure the extract bulk data operation.
4. Configure the mapper for the data extract operation and map the **schedule** parameter with the **lastProcessedDocumentID** field.
5. Drag a **Stage File** action to the integration canvas. This invokes the Configure Stage File Action wizard.
    a. On the Configure Operation page, select the **Read File in Segments** operation.
    b. Map the **filename** and **directory** from the data extract response payload in the Expression Builder:
        - XPath to **filename: $getFile/nsmpr2:GetFileResponse/nsmpr2:HcmDataExtractResponse/nsmpr2:ICSFile/nsmpr1:Properties/nsmpr1:filename**
        - XPath to **directory: $getFile/nsmpr2:GetFileResponse/nsmpr2:HcmDataExtractResponse/nsmpr2:ICSFile/nsmpr1:Properties/nsmpr1:directory**
    c. On the Schema Options page, select an existing schema from the file system.
        i. On the Format Definition page, upload the extract schema. This schema should belong to the HCM extract and must be exported from Oracle HCM Cloud.
        ii. For the **Select Repeating Batch Element** field, click the **Expression Builder** icon. The extract schema is shown in the **Source** tree.
        iii. Identify and select the repeating element from the schema.
    d. In the **Stage File** scope, use any appropriate adapter to process the extract.
6. Immediately after the **Stage File** scope, assign the processed Document ID in the schedule parameter. This enables you to use this value the next time the Oracle HCM Cloud Adapter is invoked.

Invoke an Endpoint Dynamically

You can dynamically invoke a REST endpoint/URL at runtime without configuring additional invoke connection or REST outbound details. As long as the Oracle HCM Cloud REST APIs return a response with HATEOS links, you can use this feature by mapping the HATEOS link to the invoke connection. This feature is useful in situations that require invoking a REST endpoint dynamically or when the endpoint is not known at design time. This feature is also useful in situations that require invoking multiple REST services, all of which accept the same input payload and return the same response payload as configured for the outbound endpoint. For these cases, this feature eliminates the need to create multiple connections to invoke each REST endpoint.

> **Note**
>
> Note the following restrictions.
>
> - The request and response schema must be the same as provided during endpoint configuration.
> - Template parameters are not supported while mapping these properties.
> - The HTTP verb cannot be changed for the endpoint URL. For example, if the endpoint is configured to use POST, the outgoing request uses POST even if the endpoint URI changes at runtime.
> - Because the endpoint URL is determined at runtime, there is no facility to test whether the security credentials provided during connection configuration also work with the new endpoint URL. If you think the endpoint URL determined at runtime requires a different authorization header then the original URL, you may need to provide a mapping for the authorization standard header.

This use case provides a high level overview of one way to design an integration that uses dynamic endpoints. You retrieve child objects using the REST API (for example, Primary Address is a child object of the Account parent object). The integration is designed as follows.

- An initial invoke is configured to get the Account object by using the REST API. The response of this REST API does not provide the child objects. Instead, there are HATEOS links to the child objects (that is, the Primary Address object).
- A second invoke uses the HATEOS links from the earlier response to make another invoke connection to the REST endpoint to fetch the child Primary Address object using dynamic REST endpoint support.

To change the endpoint configuration at runtime, you map one or more of the various properties under the **ConnectivityProperties** target element.

1. Create an orchestrated integration.
2. Drag an adapter into the integration canvas as an trigger connection (it can be any adapter).
3. Configure the adapter in the Adapter Endpoint Configuration Wizard.
4. Drag an initial Oracle HCM Cloud Adapter into the integration canvas as an invoke connection.
5. Configure it to use the **hcmRESTApp** service application, the **Account** object (business resource), and the **get** operation.

The response of the first invoke connection contains a collection of HATEOS links, each pointing to a child object such as **Primary Address**.

6. In the mapper between the trigger adapter connection and the Oracle HCM Cloud Adapter invoke connection, map source elements to target elements. For this example, a **PartyNumber** source element is passed to an **id** target element.
7. Add a for-each action to iterate between the HATEOS links. The value in the **Repeating Element** field is from the response object.
8. Add a switch action to get the HATEOS link corresponding to the **Primary Address** object.
9. Drag the Oracle HCM Cloud Adapter into the switch action as the second invoke connection.
10. Configure it to use the **hcmRESTApp** service application, the **Primary Address** object (business resource), and the **getAll** operation. This object uses dynamic REST endpoint support. The **Primary Address** is a collection of links. The **getAll** operation is selected for getting all the HATEOS links.
11. In the mapper immediately before the second Oracle HCM Cloud Adapter invoke connection, expand **RestApi** under **ConnectivityProperties** in the target section.
12. From the source section, map **href** to **AbsoluteEndpointURI** under **ConnectivityProperties**. The **ConnectivityProperties** schema element supports dynamic REST endpoints. The **href** element points to the **Primary Address** object link. The **href** element is invoked by the Oracle HCM Cloud Adapter.
13. If necessary, map other nodes under **ConnectivityProperties**. The runtime values provided by these mappings dynamically configure the request.

    You can also hover the cursor over these properties for brief descriptions.

Element	Description
AbsoluteEndpointURI	Represents the absolute endpoint URL that the REST Adapter invokes. Empty values are ignored. To route the request to an endpoint URL determined at runtime, provide a mapping for this element. **AbsoluteEndpointURI** takes first precedence among other URL-related properties under **ConnectivityProperties**.
BaseUri	The equivalent of the base URL provided during connection configuration. To substitute only the base URI and retain the rest of the URL, provide a mapping for this element. The mapping is ignored if **AbsoluteEndpointURI** has a nonempty runtime value.
RelativeUri	Forms the part of the endpoint URI between **BaseUri** and **?**. The mapping has no effect if **BaseUri** has an empty runtime value or **AbsoluteEndpointURI** has a nonempty runtime value. The runtime value must start with a **/**.
Uri	Use the various elements under this node to substitute runtime values for the specific parts of an endpoint URL.
Scheme	Provide a mapping to change only the scheme of the endpoint URL. Supported values are **HTTP** and **HTTPS**.

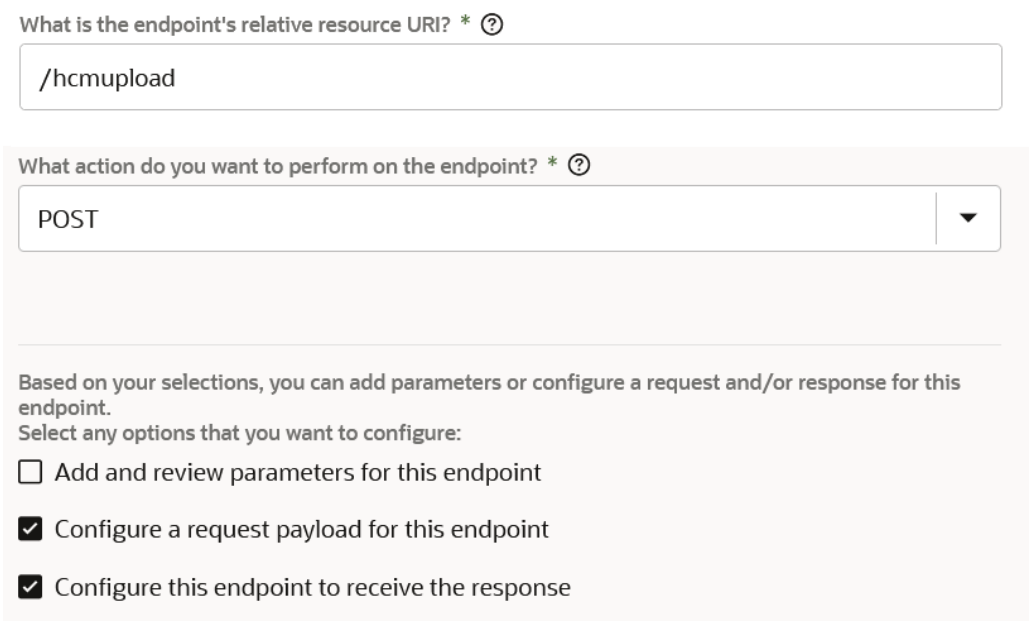
Element	Description
Host	Provide a mapping to change only the **Host** portion of the endpoint URL
Port	Provide a mapping to change only the port of the endpoint URL.
Query	Provide a mapping to change only the query portion of the endpoint URL. A query portion follows the **?**.
Path	Provide a mapping to change only the path portion of the endpoint URL. A **Path** is the part of a URI between the hostname and **?**.
Plugin	The various properties under this node impact the way the REST Adapter invokes the endpoint URL.
PostQueryString	When the runtime value is **true** and the HTTP verb is POST, the query string parameters are sent using POST as form parameters. The default is **false**.
UseFormUrlEncoding	When the runtime value is **false**, the REST Adapter uses RFC 3986-compliant encoding to encode the query parameters. The default is **true**. This is the equivalent of setting the custom header **x-ics-use-x-www-form-urlencoded** to **false**. See section “RFC 3986 Support for Encoding Query Parameters” for more information on **x-ics-use-x-www-form-urlencoded**. **x-ics-use-x-www-form-urlencoded** takes precedence when both are set.

14. Drag an FTP Adapter to the switch action for writing the **Primary Address** object response to a file on an FTP server.
15. In the mapper between the Oracle HCM Cloud Adapter and the FTP Adapter, map the **Primary Address** object details.
16. Activate and invoke the integration. The Oracle HCM Cloud Adapter invokes the endpoint URI determined at runtime.

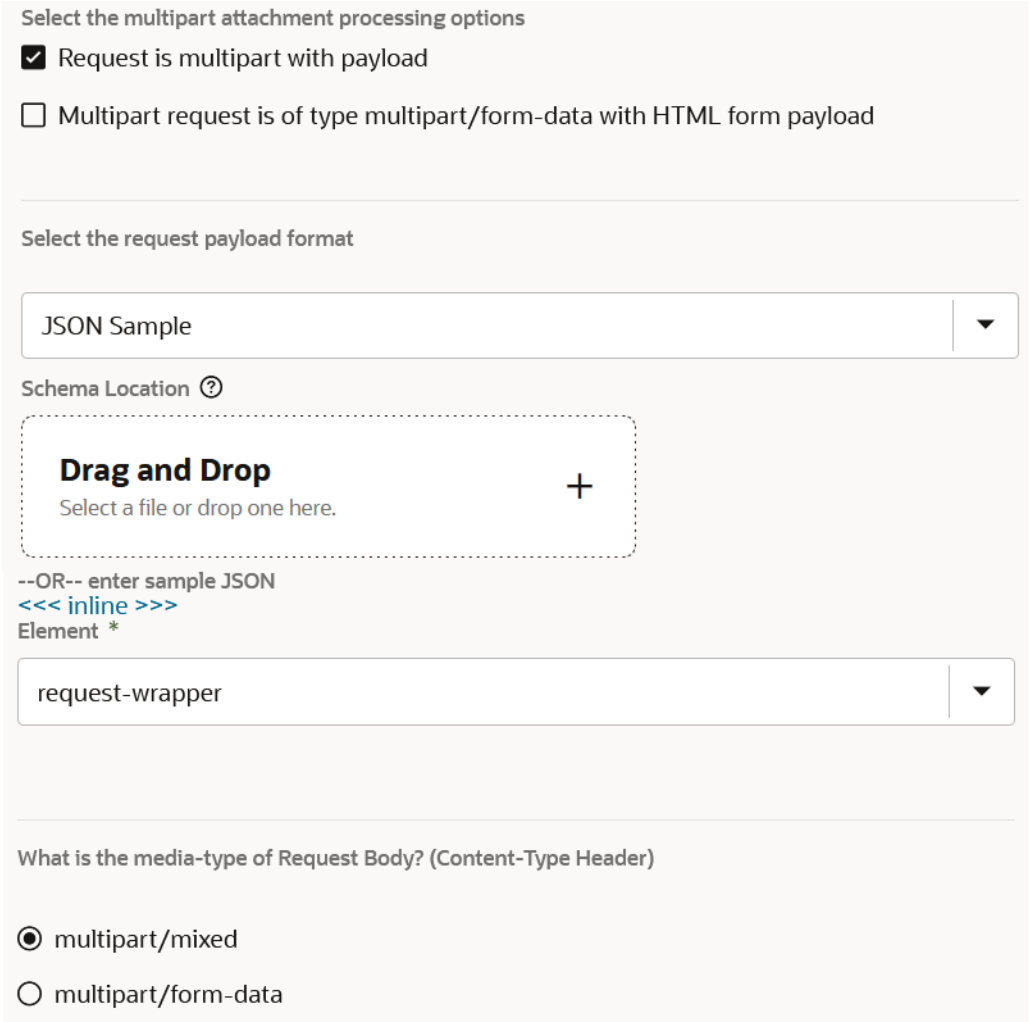
Import Bulk Data with the HCM Data Loader (HDL)

You can design an integration that imports bulk data using the HCM Data Loader. This use case provides a high-level overview.

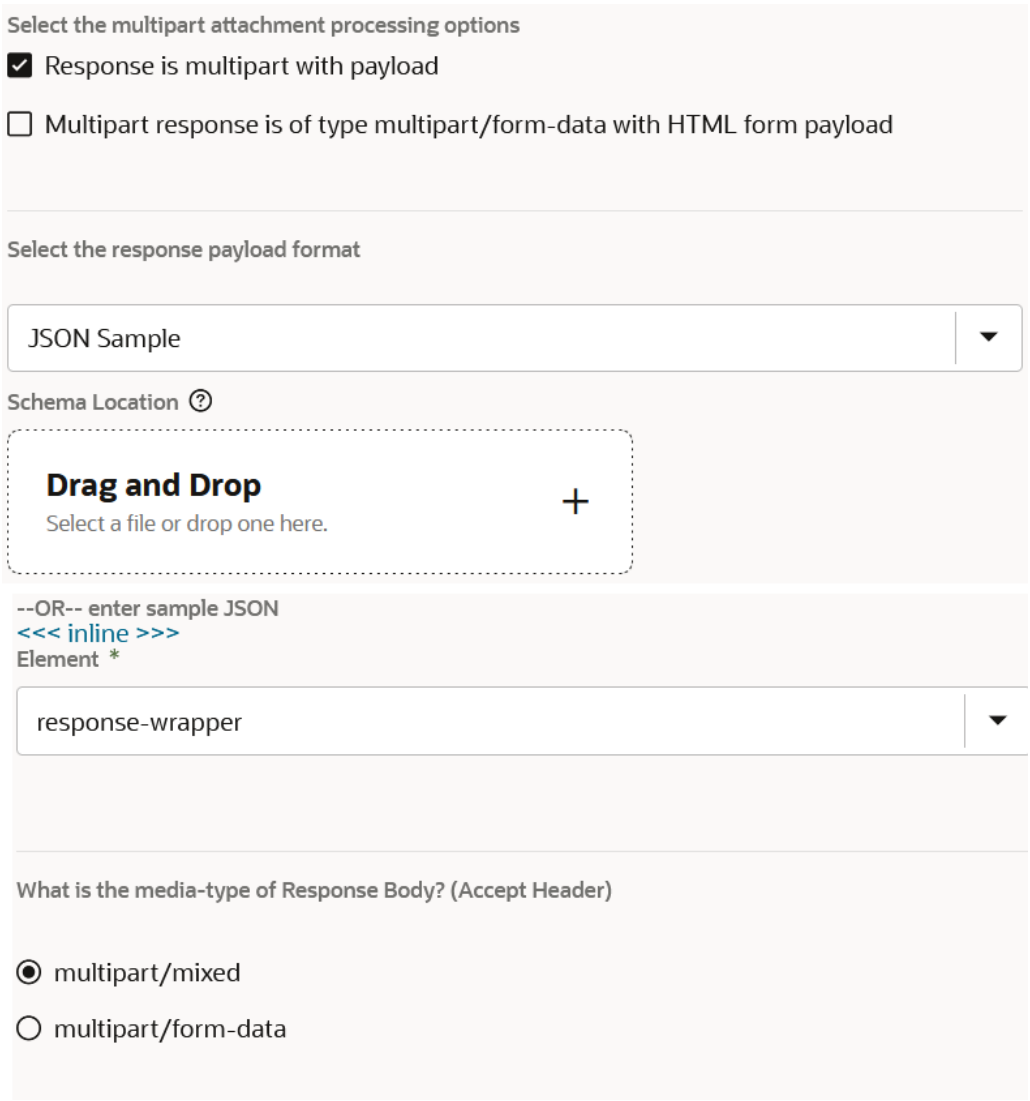
1. Create an orchestrated integration.
2. Create a REST Adapter to provide a JSON sample with Oracle HCM Cloud information.
    a. On the Resource Configuration page, specify the endpoint URI, the action to perform on the endpoint (for this example, **POST**), and any configuration options. For this example, **Configure a request payload for this endpoint** and **Configure this endpoint to receive the response** are selected.



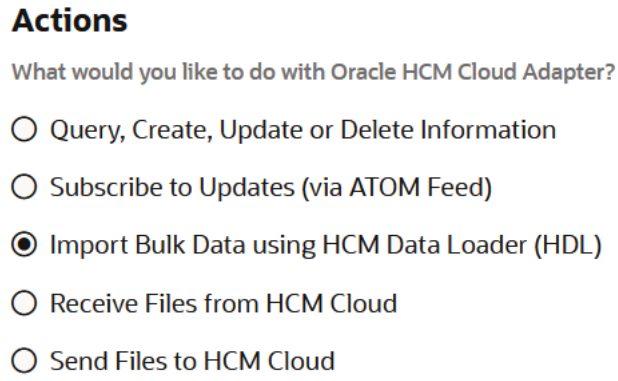
b. On the Request page, specify any multipart attachment processing options (for this example, **Request is multipart with payload** is selected), the request payload format and file (for this example, **JSON Sample**), and the media type of the request body (for this example, **multipart/mixed**d is selected).



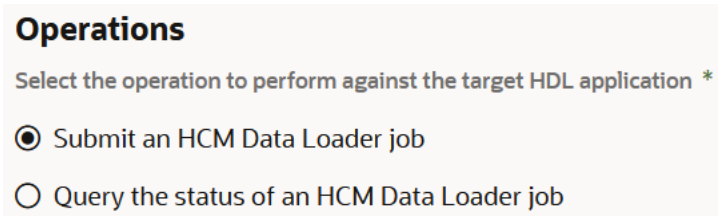
c. On the Response page, specify any multipart attachment processing options (for this example, **Response is multipart with payload** is selected), the request payload format and file (for this example, **JSON Sample**), and the media type of the request body (for this example, **multipart/mixed**d is selected).



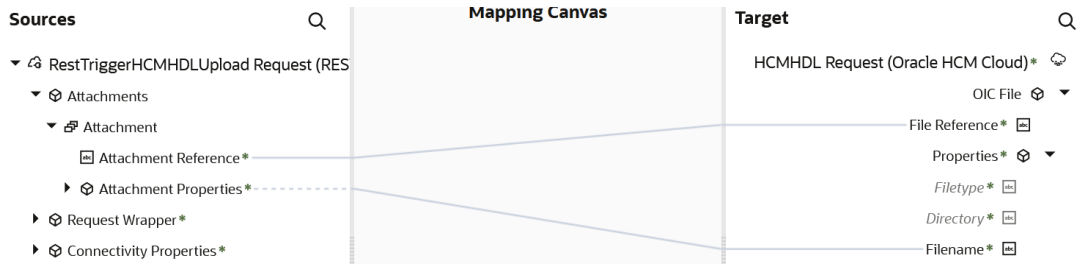
3. Add an Oracle HCM Cloud Adapter to the integration.
   a. On the Actions page, select **Import Bulk Data using HCM Data Loader (HDL)**.



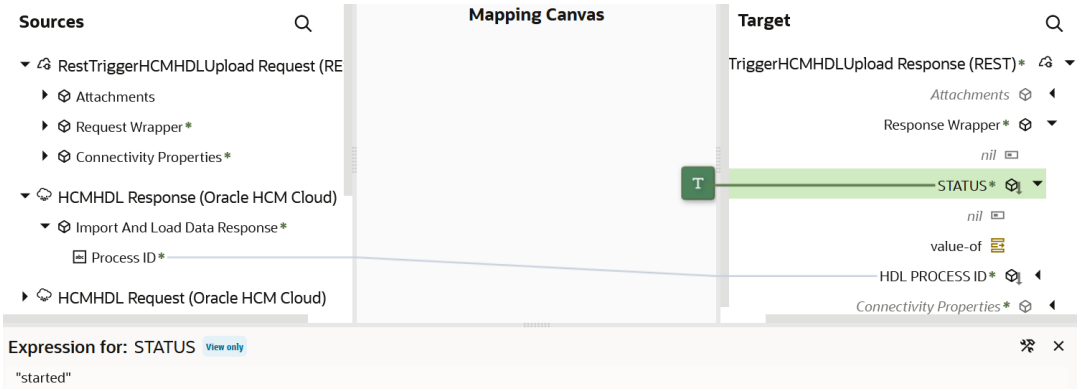
b. On the Operations page, select **Submit an HCM Data Loader job**.



4. For the request mapping, map the necessary elements from the source to the target. **Attachment Reference** is mapped to **File Reference** and **Attachment Properties** is mapped to **Filename** to send additional properties.



5. For the response mapping, map the necessary elements from the source to the target. **Process ID** is mapped to **HDL PROCESS ID**. The target **STATUS** element is set to **started**.



The completed integration looks as follows:

Select Extensible and Descriptive Flexfields in an Integration

You can select specific extensible flexfields (EFFs) and descriptive flexfields (DFFs) in the Adapter Endpoint Configuration Wizard of an Oracle HCM Cloud Adapter invoke connection. You can then map the EFFs and DFFs in the mapper.

The following use case provides an overview of how to design this type of integration.

1. Create an orchestrated integration.

2. Add a REST Adapter as a trigger connection.
3. Enter the following details:
    a. On the Basic Info page, enter a name.
    b. On the Resource Configuration page, select the **POST** action and **Configure a request payload for this endpoint** and **Configure this endpoint to receive the response**.
    c. On the Request page, select the following:
        i. Select **JSON sample** as the request payload and enter the JSON sample.
        ii. From the **Element** list, select **request-wrapper**.
        iii. For the media type of the request body, select **JSON**.
    d. On the Response page, select the following:
        i. Select **JSON sample** as the response payload and enter the JSON sample.
        ii. From the **Element** list, select **response-wrapper**.
        iii. For the media type of the response body, select **JSON**.
4. Add an Oracle HCM Cloud Adapter as an invoke connection.
5. Enter the following details.
    a. On the Basic Info page, enter a name.
    b. On the Actions page, select **Query, Create, Update, or Delete Information**.
    c. On the Operations page, select **Business (REST) Resources** from the **Browse by** list.
    d. Select an appropriate business resource and operation to perform on the resource.
    e. On the Child Resources page, select child resources with extensible or descriptive flexfields.
    f. On the Descriptive and Extensible page, select a specific flexfield and associated contexts.
6. In the request mapper between the two adapters, map appropriate source and target flexfields. For example:
7. In the response mapper after the Oracle HCM Cloud Adapter, map appropriate source and target flexfields. For example:
8. Save the integration.
9. Create business identifiers for tracking the integration during runtime.
10. Activate the integration.

Propagate OAuth User Identity Between Services

Oracle Integration provides support for OAuth identity propagation when invoking REST API operations. OAuth identity propagation enables you to securely transfer the same user identity and access credentials across services. The services involved may use the same identity domain within Oracle Integration, a different identity domain outside Oracle Integration, or a third-party identity provider.

- How Identity Propagation in Oracle Integration Works
- Propagate User Identity

How Identity Propagation in Oracle Integration Works

It is a common business requirement to propagate the identity of a user between multiple services. For example:

- You log in to a VBCS application and call Oracle Integration, which is using the identity domain in its tenancy.
- Oracle Integration then invokes an Oracle Fusion Applications endpoint, which is using a separate identity domain in a different tenancy. The Oracle Fusion Applications endpoint must know the end user making the call to drive its business logic.
- Oracle Integration also invokes a Salesforce endpoint, using a third-party identity provider outside of Oracle. The Salesforce endpoint must also know the end user making the call to drive its business logic.

Identity propagation works as follows for these types of scenarios:

- User Authentication - A user authenticates with an identity provider using their credentials.
- Token Issuance - Upon successful authentication, the identity provider issues a JWT access token containing the user's identity information and authorized scopes.
- Token propagation - When the user accesses a different service, the JWT access token is propagated with the request.
- Token validation - Each service receiving the request validates the JWT access token by verifying its integrity, expiration, and issuer.
- Getting the identity - After a successful token validation, the service extracts the user's identity from the JWT access token's claims.
- Access Controls - Based on the identity, appropriate access controls are applied to determine the user's access to resources.

Oracle Integration provides support for OAuth identity propagation with the *OAuth using the JWT User Assertion* security policy. This security policy is available if you need to use the following adapters as invoke connections to call REST API operations.

- REST Adapter
- Oracle ERP Cloud Adapter
- Oracle HCM Cloud Adapter
- Oracle CX Sales and B2B Service Adapter

No identity propagation configuration tasks are required on the Connections page or in the Adapter Endpoint Configuration Wizard for these adapters. Instead, you configure the user identity to propagate in the mapper with the **Subject** element under **Security Properties**.

> **Note**
>
> - To use identity propagation with an invoke connection created prior to Release 25.04, open the adapter in the Adapter Endpoint Configuration Wizard, click through each page, and click **Save**. These actions create the necessary **Subject** source and target elements in the mapper.
> - Identity propagation is an optional feature. If you do not want to use identity propagation with the *OAuth using JWT User Assertion* security policy, leave the **Subject** elements empty.

Propagate User Identity

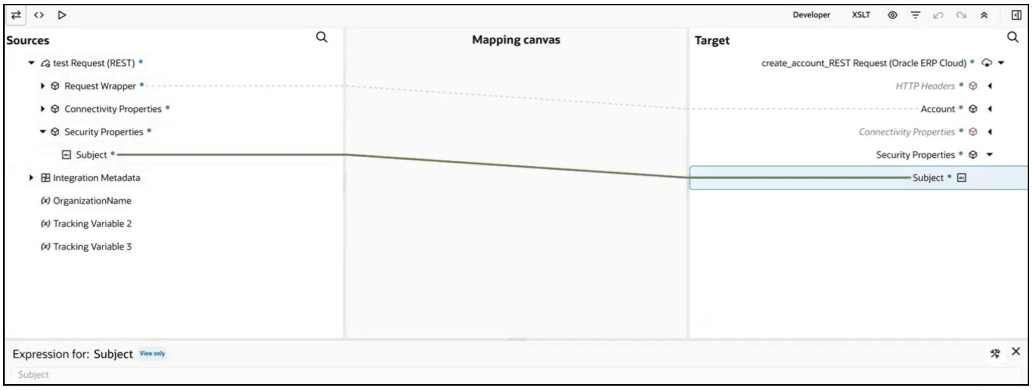
This section provides an overview of propagating user identity.

Oracle Integration and Oracle Fusion Applications are in different identity domains. For identity propagation between different identity domains to be successful, you must satisfy the following requirements:

- The user identity to propagate must be present in both the identity domain of Oracle Integration, the identity domain of Oracle Fusion Applications, and Oracle Fusion Applications Oracle Identity Management (IDM). You must have the appropriate associated roles.
- The user must have sufficient privileges to run the integration in each identity domain.

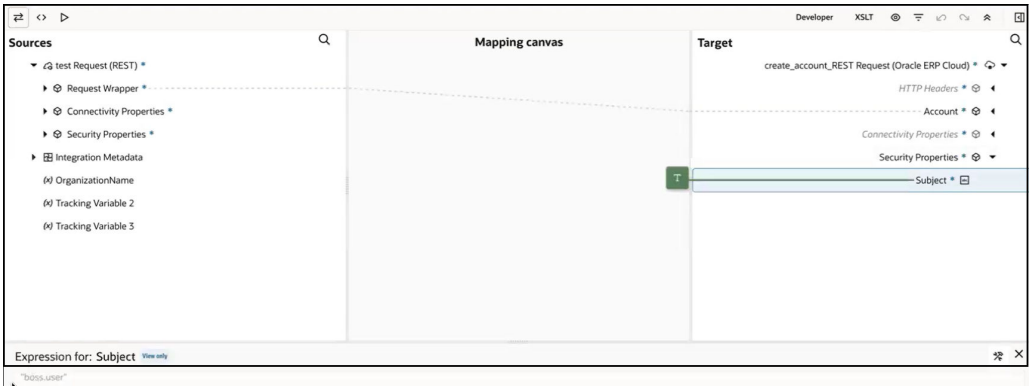
1. Create a trigger connection (for example, with the REST Adapter).
2. Create a new invoke connection with the Oracle Fusion Applications adapter you are using.
3. Configure the invoke connection to use the *OAuth using JWT User Assertion* security policy, uploading the necessary header and payload files, specifying the private key alias uploaded on the Certificates page, and specifying the scopes under **Optional security**.

4. Create an application integration.
5. Drag the trigger and invoke connections into the integration canvas for configuration with the Adapter Endpoint Configuration Wizard.
6. Open the mapper.
7. Set the user that has permission to execute the integration through either of two options in the mapper:
   - Expand **Security Properties** in the **Sources** and **Target** areas and map the user in the source **Subject** element to the target **Subject** element.



- Expand **Security Properties** in the **Target** area and manually set the user for the **Subject** element in the Expression Builder. For this example, this option is demonstrated. The user name is specified in the Expression Builder as `"boss.user"`.

> **Note**
>
> If you run the integration in a tool such as `Postman`, you use a user name/password or client credentials. The username is populated in the subject node on the trigger side and is used to get the JWT access token.



8. Complete design of your integration.
9. Activate and run the integration as the `boss.user` user specified in the mapper. The activity stream indicates the run was successful.

   If the integration run was unsuccessful, the following errors may have occurred.

Error	Reason for Error
`The 401 error message is usually returned by services that require user credentials. So if you have got this error then it probably means that you entered an invalid username or password.`	If the user is present in both the identity domain of Oracle Integration and the identity domain of Oracle Fusion Applications, and the Oracle Fusion Applications user can receive the token correctly, but that user lacks sufficient privileges to run the integration.
`Request to access token failed. Cause: status = 400 Error: {\"error\":\"invalid_grant\",\"error_description\":\"Invalid user assertion: The user name that you entered is invalid. Contact your system administrator.\"`	If the user is present in the identity domain of Oracle Integration, but not present in the identity domain of Oracle Fusion Applications, a token cannot be generated.

Watch a video to learn more:

Video

5

Troubleshoot the Oracle HCM Cloud Adapter

Review the following topics to learn about troubleshooting issues with Oracle HCM Cloud Adapter.

Topics:

Extraction of Emps Business Objects from Oracle HCM Cloud Requires a Service Request

If you use the getall REST service to extract data from your Oracle HCM Cloud environment, the Oracle HCM Cloud Adapter is populated with various business objects, but not the Emps business object. This occurs because the Emps resource is currently under controlled availability. The following role is also required to populate Emps business objects:

Use REST Service - Employees

PER_REST_SERVICE_ACCESS_EMPS_PRIV

Open a service request through Oracle Support Services to gain access to the Emps business object.

The function security privilege required for ATOM feeds is:

Use ATOM Feeds - Employees Workspace

PER_ATOM_WORKSPACE_ACCESS_EMPLOYEES_PRIV.

Avoid Missing Atom Entries When Processing Them Page-Wise

The Oracle HCM Cloud Adapter fetches fewer entries from the Atom server as compared to Postman or `curl` in a single invoke. The Oracle HCM Cloud Adapter provides a solution to avoid missing these entries when processing them page-wise.

Assume you fetch a single page of entries when polling an Atom feed with Postman. If the last entry in the response/page has an identical updated timestamp as the next few entries on the consecutive invoke's response/page, these further entries (with the same updated timestamp) are never included in the subsequent response because the `updated-min` parameter is not inclusive. Similarly, in case future dated updates need to be fetched immediately and the published timestamp needs to be used, if the last entry in the response/page has an identical published timestamp as the next few entries on the consecutive invoke's response/page, these

further entries (with the same published timestamp) are never included in the subsequent response because the `published-min` parameter is not inclusive.

To fix this issue, the Oracle HCM Cloud Adapter processes `N-` entries (the number of entries with the same updated timestamp/published timestamp towards the end of the page). Here, `N` is the total number of entries on the page. The remaining entries are included in the next invocation. This ensures that no entries are skipped.

Unsupported SOAP APIs Available for Selection

When browsing for services on the Operations page of the Adapter Endpoint Configuration Wizard, unsupported services may appear for selection. If you select an unsupported service (for example, the CreateWorker service), a null pointer exception is displayed.

See Chapter Business Object Service Structure in *SOAP Web Services for HCM* for the only supported SOAP APIs.

ATOM Feeds Option Not Appearing for Selection in the Operations Page

If the **Subscribe to Updates (via ATOM Feed)** option does not appear for selection on the Actions page in the Adapter Endpoint Configuration Wizard, note the following potential causes:

- The interface catalog URL has not been specified on the Connections page.
- The Atom feed feature may not be enabled for your Oracle Integration instance. Contact Oracle Support Services.

See Configure Connection Properties.

Manual Metadata Refresh is Required if Updating the Connection to Use the Interface Catalog URL

If an existing Oracle HCM Cloud Adapter connection is updated to add the interface catalog URL, you must manually refresh the metadata. Go to the Connections page and find the Oracle HCM Cloud Adapter to refresh.

See Refresh Integration Metadata.