Oracle® Cloud

Using the Oracle CX Sales and B2B Service Adapter with Oracle Integration Generation 2

E85494-47
October 2024

Oracle Cloud Using the Oracle CX Sales and B2B Service Adapter with Oracle Integration Generation 2,

E85494-47

Primary Author: Oracle Corporation

Contents

Preface

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

> **Note:**
>
> The use of this adapter may differ depending on the features you have, or whether your instance was provisioned using Standard or Enterprise edition. These differences are noted throughout this guide.

Topics:

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.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also

mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Related Resources

See these Oracle resources:

- Oracle Cloud

  `http://cloud.oracle.com`

- *Using Integrations in Oracle Integration Generation 2*
- *Using the Oracle Mapper with Oracle Integration Generation 2*

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 CX Sales and B2B Service Adapter

Review the following conceptual topics to learn about the Oracle CX Sales and B2B Service 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 CX Sales and B2B Service Adapter Capabilities

The Oracle CX Sales and B2B Service Adapter enables modern selling with tools that are easy to deploy and use, completely mobile, packed with powerful analytics, and built for collaborative selling and revenue generation. The Oracle CX Sales and B2B Service (formerly called Oracle Engagement Cloud) includes a set of features for creating and tracking sales campaigns, developing leads into business opportunities, and pursuing opportunities to generate revenue. Sales accounts, leads, and opportunities can be automatically assigned to territories and sales teams.

> **Note:**
>
> - Oracle Engagement Cloud is now known as Oracle CX Sales and B2B Service. Use the Oracle CX Sales and B2B Service Adapter to create an integration with Oracle CX Sales and B2B Service.
> - Ensure that you have reviewed the Oracle HCM Cloud Adapter Capabilities for supported SOAP services in Oracle HCM Cloud.

The Oracle CX Sales and B2B Service Adapter enables you to create an integration with an Oracle CX Sales and B2B Service application.

Oracle CX Sales and B2B Service Adapter enables customers to easily integrate their on-premises or SaaS applications with the Oracle CX Sales and B2B Service without having to know the specific details involved in the integration.

The Oracle CX Sales and B2B Service Adapter provides the following capabilities:

- A WSDL that defines strongly-typed message structures (request and response types) for the selected objects and the name of operations. It provides a simplified user experience

for creating data mappings and accessing Oracle CX Sales and B2B Service-related business objects/resources-specific elements through other Oracle Integration activities at design time while constructing integrations with Oracle Fusion Application services.

- Provides declarative support for subscribing to business events raised by various modules in the Oracle CX Sales and B2B Service and Oracle Supply Chain Cloud. See Supported Business Events.
- Simplified connection creation: Automatically identifies the required service catalog service WSDL, optional event catalog URL, and optional interface catalog URL to use based on the Oracle CX Sales and B2B Service host name you specify when creating a new connection on the Connections page.
- Generates automatic mapping to the exposed business object, event, or Oracle Fusion Applications REST API resource that you select during adapter configuration:
  - 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.
  - Event: Represents an event document to which you subscribe. The event is raised by the Oracle CX Sales and B2B Service application. See Supported Business Events.

    You can also create custom business events in the Oracle CX Sales and B2B Service that can be published and subscribed to with the Oracle CX Sales and B2B Service Adapter. See Enable Event Subscriptions in the Oracle CX Sales and B2B Service.
  - Business (REST) API: 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.

    > **Note:**
    >
    > The Oracle CX Sales and B2B Service Adapter currently pulls in all resources exposed by the interface catalog and displays them for selection. See Resource Types in *REST API for CX Sales and B2B Service*.

- Enables you to view annotations on Oracle CX Sales and B2B Service Adapter elements in the mapper. See About Mappings in *Using the Oracle Mapper with Oracle Integration Generation 2*.
- 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.
- Automatically handles security policy details required to connect to the Oracle CX Sales and B2B Service application.
- Supports the following security policies for selection during Oracle CX Sales and B2B Service Adapter connection configuration:
  - Username Password Token
  - OAuth Authorization Code Credentials
- Provides standard error handling capabilities.

- Enables you to map business objects that have polymorphic data structures.
- Supports the token-based authentication scheme for business and FBDI event messages originating from Oracle Fusion Applications.
  Oracle CX Sales and B2B Service Adapter token-based authentication uses time-sensitive session tokens generated using strong encryption on the Oracle Fusion Applications event handler and passed to Oracle Integration. The session token is validated by Oracle Integration against the Oracle CX Sales and B2B Service application sender and added to local cache. Tokens are never persisted on Oracle Fusion Applications or Oracle Integration. Token-based authentication provides the following benefits:
  - The token refreshes automatically.
  - You don’t have to worry about expiration.
  - It's difficult to compromise because tokens are self-generated.
  - No additional settings are required.

Support for Calling Integrations with Concrete Values of Custom Objects from the Groovy Script Editor

The Oracle CX Sales and B2B Service Adapter inbound endpoint WSDL interface supports the use of concrete values for custom business objects in the WSDL in place of `xsd:anyType` parameters. This feature enables you to use the groovy script editor to create scripts to invoke integrations. The groovy script editor is unable to recognize `xsd:anyType` parameters.

See Integrate Groovy Scripts.

Supported Business Events

You can subscribe to the following business events when configuring the Oracle CX Sales and B2B Service Adapter as a trigger (inbound) connection in the Adapter Endpoint Configuration Wizard.

The business events in the Oracle CX Sales and B2B Service also handle all child events. For example, if an Opportunity is created first, it raises an Opportunity Created event and next a revenue line item is added to that Opportunity the next day. The event is then an Opportunity Updated event, but the final payload includes the revenue line item. That is, specific child events are not created for the child objects that have a relationship to the parent object.

Business Event	Available with Release
Account Created Event	10 and later
Account Updated Event	10 and later
Account Deleted Event	10 and later
Activity Created Event	13
Activity Updated Event	13
Activity Deleted Event	13
Activity Note Created Event	13
Activity Note Updated Event	13
Activity Note Deleted Event	13
Asset Created Event	13
Asset Updated Event	13
Asset Deleted Event	13

Business Event	Available with Release
BusinessPlan Created Event	13
BusinessPlan Updated Event	13
BusinessPlan Deleted Event	13
Business Plan Note Created Event	13
Business Plan Note Updated Event	13
Business Plan Note Deleted Event	13
Contact Created Event	10 and later
Contact Updated Event	10 and later
Contact Deleted Event	10 and later
Contract Status Changed Event	13
Deal Created Event	11 and later
Deal Updated Event	11 and later
Deal Note Created Event	13
Deal Note Updated Event	13
Deal Note Deleted Event	13
Lead Created Event	11 and later
Lead Updated Event	11 and later
Lead Deleted Event	11 and later
Lead Note Created Event	13
Lead Note Updated Event	13
Lead Note Deleted Event	13
Loyalty Member Created Event	13, 17D through 18C
Loyalty Member Updated Event	13, 17D through 18C
Loyalty Member Voucher Created Event	13, 17D through 18C
Loyalty Member Voucher Updated Event	13, 17D through 18C
Loyalty Member Card Created Event	18B and 18C
Loyalty Member Card Updated Event	18B and 18C
Loyalty Member Dynamic Attribute Created Event	18C
Loyalty Member Dynamic Attribute Updated Event	18C
Loyalty Member Promotion Enrollment Event	18C
Loyalty Member Referral Event	13, 18B, 18C
Loyalty Member Tier Change Event	18C
Loyalty Transaction Created Event	18B and 18C
Loyalty Transaction Updated Event	18C
Opportunity Created Event	11 and later
Opportunity Updated Event	11 and later
Opportunity Deleted Event	11 and later
Opportunity Note Created Event	13
Opportunity Note Updated Event	13
Opportunity Note Deleted Event	13
Partner Created Event	11 and later

Business Event	Available with Release
Partner Updated Event	11 and later
Partner Contact Created Event	11 and later
Partner Contact Updated Event	11 and later
Partner Note Created Event	13
Partner Note Updated Event	13
Partner Note Deleted Event	13
Recurrence Activity Created Event	13
Recurrence Activity Updated Event	13
Recurrence Activity Deleted Event	13
Resolution Request Updated	13
Service Request Created Event	13
Service Request Updated Event	13
Service Request Selected Attribute value Changed Event	13
Service Request Deleted Event	13
Service Request Message Created Event	13
Work Order Created Event	13
Work Order Updated Event	13

Oracle CX Sales and B2B Service Adapter Restrictions

Note the following Oracle CX Sales and B2B Service restrictions.

- Oracle Fusion Applications allows clients to access the public event catalog using the HTTP basic authentication scheme. When the client is not allowed to communicate with the catalog using this scheme, they receive the following error: `Server redirected too many times (20)`. This occurs while testing the Oracle Cloud connection. You must file a service request with Oracle Fusion Applications to resolve this issue.

> **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.

See Connectivity Certification Matrix.

Oracle CX Sales and B2B Service Adapter Use Cases

Common use cases for the Oracle CX Sales and B2B Service Adapter are as follows:

- Customer 360 degree view across service, sales, and marketing
- Leads management with marketing
- Opportunity to order with ERP systems
- Real-time synchronization of incidents and activities with services

Workflow to Create and Add an Oracle CX Sales and B2B Service 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.

This table lists the workflow steps for both adapter tasks and overall integration tasks, and provides links to instructions for each step.

Step	Description	More Information
1	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 CX Sales and B2B Service Adapter Connection
2	Create the integration. When you do this, you add trigger (source) and invoke (target) connections to the integration.	Create Integrations and Add the Oracle CX Sales and B2B Service Adapter to an Integration
3	Map data between the trigger connection data structure and the invoke connection data structure.	Map Data in *Using Integrations in Oracle Integration Generation 2*
4	(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 Generation 2*
5	Activate the integration. **Note**: If you deactivate an integration that contains a business event subscription, a message is displayed asking if you want to delete the event subscription. See Deactivate an Integration with Business Events.	Manage Integrations in *Using Integrations in Oracle Integration Generation 2*
6	Monitor the integration on the dashboard.	Monitor Integrations in *Using Integrations in Oracle Integration Generation 2*
7	Track payload fields in messages during runtime.	Assign Business Identifiers for Tracking Fields in Messages and Manage Business Identifiers for Tracking Fields in Messages in *Using Integrations in Oracle Integration Generation 2*
8	Manage errors at the integration level, connection level, or specific integration instance level.	Manage Errors in *Using Integrations in Oracle Integration Generation 2*

2

Create an Oracle CX Sales and B2B Service 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 CX Sales and B2B Service Adapter.

Subscribe to the Oracle CX Sales and B2B Service

This action enables you to create an Oracle CX Sales and B2B Service user account with the correct privileges. You specify this user account when creating an Oracle CX Sales and B2B Service Adapter connection on the Connections page.

Enable Event Subscriptions in the Oracle CX Sales and B2B Service

Before you can subscribe to events with the Oracle CX Sales and B2B Service Adapter, you must perform a series of configuration tasks. For this example, Oracle Service Cloud (RightNow) Adapter is the other connection with the Oracle CX Sales and B2B Service in this integration.

- Enable the Trading Community Events Profile Option
- Create Custom Business Events

Verify the Source System Record in the Oracle CX Sales and B2B Service

The integration is designed to work with the Oracle CX Sales and B2B Service Release 10.

To verify the source system record in the Oracle CX Sales and B2B Service:

1. Log in to an Oracle CX Sales and B2B Service instance with a user with system administrator privileges.
2. Navigate to the Setup and Maintenance page.
3. Select the **All Tasks** tab.
4. Enter `Manage Trading community Source System` in the **Name** field, and click **Search**.
5. In the **Search Results** section, click the icon under **Go to Task**.

   The Manage Trading Community Source Systems page is displayed.
6. Select **Starts with** from the **Code** drop-down menu.
7. To the right of **Starts with**, enter a value in the field, then click **Search**. For this example, `RNOW` is entered.
8. In the **Search Results** section, verify the value you entered (for this example, **RNOW**) is displayed in the **Code** column and ensure the **Enable for Trading Community Members** checkbox is selected. If the **Enable for Trading Community Members** check box is not selected, then perform the following steps:
   a. Select the **RNOW** row.
   b. Click the **Edit** icon.
   c. Select the **Enable for Trading Community Members** checkbox.

Create the Source System Record in the Oracle CX Sales and B2B Service

To create the Source System Record in the Oracle CX Sales and B2B Service:

If the Source System definition RNOW record is not found in the Oracle CX Sales and B2B Service instance, then follow the steps below to create one:

1. Log in to the Oracle CX Sales and B2B Service instance with a user with system administrator privileges.
2. Navigate to the Setup and Maintenance page.
3. Select the **All Tasks** tab.
4. Enter `Manage Trading community Source System` in the **Name** field, and click **Search**.
5. In the **Search Results** section, click the icon under **Go to Task**.

   The Manage Trading Community Source Systems page is displayed.
6. Select **Starts with** from the **Code** drop-down menu.
7. Enter `RNOW` in the field, then click **Search**. For this example, Oracle Service Cloud (RightNow) Adapter is the other connection with which the Oracle CX Sales and B2B Service is communicating.
8. Under **Search Results**, click the **New** icon.

9. On the Create Source System page, fill in the values as follows. For this example, Oracle Service Cloud (RightNow) Adapter is the other connection with which the Oracle CX Sales and B2B Service is communicating.
    - **Code** field: Enter `RNOW`.
    - **Name** field: Enter `Service Cloud`.
    - **Description**: field: Enter a description. For example:

      ```
      Maintains cross references between the Oracle Fusion Applications
      database and records imported using comma-separated files.
      ```

    - **Enable for Trading Community Members** checkbox: Select the checkbox.
10. Click **Save**, then click **Close**.

Verify Source System Entities

To verify source system entities:

1. Navigate to the Setup and Maintenance page.
2. Select the **All Tasks** tab.
3. Enter `Manage Source System Entities` in the **Name** field, and click **Search**.
4. In the **Search Results** section, click the icon under **Go to Task**.

   The Manage Source System Entities page is displayed.
5. From the **Source Systems for Trading Community Members** list, select **Service Cloud (RightNow)**. For this example, Oracle Service Cloud (RightNow) Adapter is the other connection with which the Oracle CX Sales and B2B Service is communicating.
6. In the **Service Cloud (RightNow): Entities** section, ensure that the **Address**, **Contact Points**, and **Parties** checkboxes are selected.

Enable the Trading Community Events Profile Option

This profile option only applicable to Trading Community Architecture (TCA) objects such as Accounts and Contacts. If the profile option is set to **No** out of the box, the events for the Accounts and Contacts objects are disabled. The rest of the objects do not require any profile option and are enabled to raise business events out of the box.

To enable the trading community events profile option:

1. Navigate to the Setup and Maintenance page.
2. Select the **All Tasks** tab.
3. Enter `Manage Trading Community Common Profile Options` in the **Name** field, and click **Search**.
4. In the **Search Results** section, click the icon under **Go to Task**.

   The Manage Trading Community Common Profile Options page is displayed.
5. Select **HZ_ENABLE_EVENT_TRACKING**.

   The Manage Trading Community Common Profile Options page for the HZ_ENABLE_EVENT_TRACKING page is displayed.
6. In the **HZ_ENABLE_EVENT_TRACKING:Profile Values** section, set the **Site** level **Profile Value** to **Yes**.

7. Click **Save**, then click **Close**.
8. On the Manage Trading Community Common Profile Options page, select **HZ_INVOKE_OBJ_WF_ON_TRACKING**.

   The Manage Trading Community Common Profile Options page for the HZ_INVOKE_OBJ_WF_ON_TRACKING page is displayed.
9. In the **HZ_INVOKE_OBJ_WF_ON_TRACKING: Profile Values** section, set the **Site** level **Profile Value** to **Yes**.
10. Click **Save**, then click **Close**.

Create Custom Business Events

You can create custom business events in Application Composer that are visible for selection when configuring the Oracle CX Sales and B2B Service Adapter as a trigger connection in the Adapter Endpoint Configuration Wizard. You must access Application Composer through the Oracle Fusion Applications user interface. Select **CRM Cloud** from the **Applications** list in Application Composer to create the custom objects and promote them as custom events to be consumed by the Oracle CX Sales and B2B Service Adapter.

See technical note 2535444.1 at My Oracle Support for instructions.

Create an Integration User Account

To invoke an Oracle CX Sales and B2B Service service catalog or event catalog web service from Oracle Integration, you create a separate user.

To create the integration user account:

1. Log in to the Oracle CX Sales and B2B Service with a user with system administrator privileges.
2. Navigate to **Navigator** > **My Team** > **Manage Users**.

   The Manage Users page is displayed.
3. Click **Manage Users**.
4. Click the **Create New User** icon beside **Show Photo**.
5. Enter the following information, and click **Save**.

Field	Description
Last Name	Enter `CUSTOMER_OIC_INTEG_USER`
Email	Enter a valid email address.
Hire Date	Enter the date.
User Name	Enter `CUSTOMER_OIC_INTEG_USER`.
Person Type	Enter `Employee`.
Legal Employer	Select a valid legal organization.
Business Unit	Select a valid business unit.
Send user name and password	Select this checkbox.
User Login	Enter `CUSTOMER_OIC_INTEG_USER`.
Password	Enter a password for the username.

A notification email is sent to the email address after the user is created.

6. Log out of the Oracle CX Sales and B2B Service.
7. Log in to the Oracle CX Sales and B2B Service instance with `CUSTOMER_OIC_INTEG_USER` and the temporary password provided in the notification email.
8. Change the password after logging in initially.

   The Oracle CX Sales and B2B Service page appears.
9. Log out of the Oracle CX Sales and B2B Service.

Assign Integration Roles

You associate a user with roles and privileges in Oracle Authorization Policy Manager on the Oracle Entitlements Server for Releases 10 and 11. If using Releases 12 and later, you use the Security Console to manage users and roles.

You can configure an Oracle Integration instance to use the Username Password Token security policy to access the resources in an Oracle CX Sales and B2B Service Adapter instance.

An Oracle CX Sales and B2B Service Adapter instance exposes service catalogs and event catalogs to Oracle Integration. These resources are secured in the Oracle CX Sales and B2B Service Adapter. You need to assign the required roles and privileges to a user.

> **Note:**
>
> For prebuilt integrations, you must use the name **CUSTOMER_OIC_INTEG_USER**. In prebuilt integrations, connections and filter expressions for echo suppression refer to **CUSTOMER_OIC_INTEG_USER** as the user name. For new integrations, the user name can be anything.

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.
Customer Relationship Management Application Administrator	This role is supported in releases 12 and later.
SOAOperator	This role is required to receive business events.
FND_MANAGE_CATALOG_SERVICE_PRIV	Role for managing the web services catalog.

Additional roles may be required per each interface requirement.

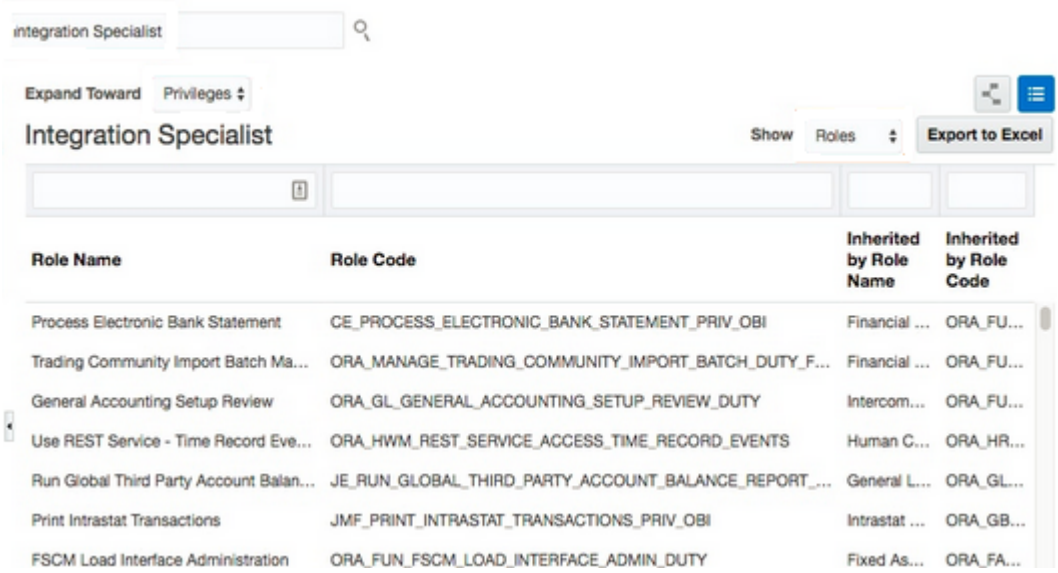
> **Note:**
>
> You must have administrator privileges in Oracle Authorization Policy Manager to perform the following steps.

Assigning Integration Roles in Releases 12 and Later

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**.

For example, the Roles page for the **Integration Specialist** user looks as follows:



See Oracle Cloud Applications Security Console of *Securing Sales and Service*.

Assigning Integration Roles in Releases 10 and 11

1. Log in to Oracle Authorization Policy Manager.
2. Under **Search**, select `Users` from the **For** dropdown list.
3. Enter `CUSTOMER_OIC_INTEG_USER` (for prebuilt integrations) or any name (for new integrations) in the **Search** field, then click the **Search** icon.
4. Select that name in the search results, then click the **View** icon.

5. On the ***username*** tab (**CUSTOMER_OIC_INTEG_USER** or whatever name you created), click the **Application Role Assignments** subtab.
6. Click **Map**.
7. In the Map Application Roles to User dialog:
    a. Select **crm** from the **Application Name** dropdown list.
    b. Select **Contains** from the **Role Name** dropdown list, enter `ALL_INTEGRATION_POINTS_ALL_DATA` (for releases 11 and earlier) or `Customer Relationship Management Application Administrator` (for releases 12 and later) in the **Role Name** field, and then click **Search**.
    c. Select **ALL_INTEGRATION_POINTS_ALL_DATA** (for releases 11 and earlier) or **Customer Relationship Management Application Administrator** (for releases 12 and later) in the search results, then click **Map Roles**.

       The role is mapped for the user under **crm**.
8. Click **Map**.
9. Perform the following steps in the Map Application Roles to User dialog.
    a. Select **hcm** from the **Application Name** dropdown list, and repeat steps 7.b, and 7.c.
    b. The role is mapped for the user under **hcm**.
10. Click **Map**.
11. Perform the following steps in the Map Application Roles to User dialog.
    a. Select **fscm** from the **Application Name** dropdown list, and repeat steps 7.b, 7.c.

       The role is mapped for the user under **fscm**.
12. Click **Map**.
13. Perform the following steps in the Map Application Roles to User dialog.
    a. Select **soa-infra** from the **Application Name** dropdown list.
    b. Select **Contains** from the dropdown menu next to **Display Name**.
    c. Enter `SOA Operator` in the **Display Name** field, and then click **Search**.
    d. Select the **SOA Operator** role in the search results, then click **Map Roles**.

       The role is mapped to the user under **soa-infra**.
14. On the ***username*** tab (**CUSTOMER_OIC_INTEG_USER** or whatever name you created), click **Find Policies** at the top-right corner of the page
15. In the Choose an Application dialog, select **fscm** and click **OK**.

    The **Search Authorization Policies** tab is displayed.
16. Click the **New** icon under **Functional Policies** on the Search Authorization Policies page.
17. On the **Untitled** tab, enter `Policy for manage services catalog` in the **Name** field.
18. Ensure your username (**CUSTOMER_OIC_INTEG_USER** or whatever name you created) is listed under **Principals**. If the use is not listed under **Principals**, then click **+** next to **Principals**.
19. Perform the following steps in the Search Principal dialog.
    a. Select the **Users** tab.

b. Select **Starts With** from the **User Name** dropdown list, enter the user name (`CUSTOMER_OIC_INTEG_USER` or whatever name you created) in the **User Name** field, and then click **Search**.

c. Select the user name in the search results, then click **Add Selected**.

d. Click **Add Principals**.

20. Select **All** next to **Match under Principals**.
21. Click **+** next to **Targets**.
22. In the Search Target dialog, click the **Entitlements** tab, then select **Starts With** from the **Name** dropdown list.
23. Enter `FND_MANAGE_CATALOG_SERVICE_PRIV`, then click **Search**.
24. Select **Manage Webservices catalog** in the search results, then click **Add Selected**.
25. Ensure **Manage Webservices catalog** is added onto **Selected Targets**.
26. Click **Add Targets**.

    A new Policy for manage services catalog is added to the user.
27. Click **Save** on the **Untitled** tab.
28. Sign out from Oracle Authorization Policy Manager.

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

Check if you have enabled Location-Based Access Control (LBAC) for Fusion Applications (for the Oracle CX Sales and B2B Service).

If LBAC is enabled, you must allowlist (explicitly allow identified entities access) the Oracle Integration NAT Gateway IP address in your LBAC.

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.

If you do not perform this task, you can receive a `401 Access Denied` error or `403 Forbidden` error from Oracle Fusion Applications.

Upload a Security Certificate

Certificates validate outbound SSL connections. If you make an SSL connection in which the root certificate does not exist in Oracle Integration, an exception is thrown. In that case, you must upload the appropriate certificate. A certificate enables Oracle Integration to connect with external services.

See Upload an SSL Certificate.

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, a non-Oracle Fusion

Applications identity domain (for example, the Oracle Integration identity domain), or Oracle Identity Cloud Service.

Topics:

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

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 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**.
2. a. On the Details page, provide a name (for example, `FA Resource`), and click **Next**.
    b. On the Client page, click **Next** without making changes.
    c. On the Resources page, click **Configure this application as a resource server now**.
    d. Optionally update the value in the **Access Token Expiration** field.
    e. Select **Is Refresh Token Allowed**.

f. 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`

g. In the **Scopes** section, click **Add**.

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

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

j. Select **Requires Consent**.

k. Click **Add**, then click **Next**.

l. On the Web Tier Policy and Authorization pages, click **Next** without making any changes.

m. Click **Finish** to complete resource application creation.

n. Click **Activate** to activate your client application. 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 Infrastructure 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 **Next**.
11. In the **Client configuration** box, select **Configure this application as a client now**.
12. For authorization code, select **Refresh token** and **Authorization code** in the **Allowed grant types** section.
13. 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 Generation 2 to Oracle Integration Generation 2.
> - 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 Generation 2 instances	Yes.	`https://OIC_instance_URL.region.ocp.oraclecloud.com/icsapis/agent/oauth/callback`
Created on instances upgraded from Oracle Integration Generation 2 Generation 2 to Oracle Integration Generation 2	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.

14. Under **Resources**, 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.
15. Search for the Oracle Fusion Applications resource application created in Create an Identity Domain Resource Application to Represent the Oracle Fusion Applications Resource.
16. Select the resource and click **>**.
17. Select the scope, then click **Add**.
18. Click **Next** without making changes on the Resource and Web Tier Policy pages.
19. On the Authorization page, click **Finish**.
The Application Added dialog 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. Activate the application.

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.

- 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 All REST Endpoints in *REST API for Oracle Identity Cloud Service*.
   b. Copy the endpoint.
   c. Get the identity domain URL from the Oracle Cloud Infrastructure 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 payloads of the certificates. There are two sections in the payload:

- Identity domain certificate
- Certificate authority (CA) certificate

Examples of the type of response you receive are provided. See Retrieve the Tenant's Signing Certificate in JWK Format.

f. Copy both certificate 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_certificate
. . .
. . .
-----END CERTIFICATE-----
```

You can validate the certificate. For example:

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

2. File a service request (SR) with Oracle Fusion Applications and attach your certificates for upload. You cannot upload the certifications 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. On the Details page, provide a name (for example, `FA Resource`), and click **Next**.
   j. On the Client page, click **Next** without making changes.
   k. On the Resources page, click **Configure this application as a resource server now**.
   l. Optionally update the value in the **Access Token Expiration** field.
   m. Select **Is Refresh Token Allowed**.
   n. 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`

   o. In the **Scopes** section, click **Add**.
   p. In the **Scope** field, enter `/`.
   q. In the **Description** field, enter **All**.
   r. Select **Requires Consent**.

s. Click **Add**, then click **Next**.

t. On the Web Tier Policy and Authorization pages, click **Next** without making any changes.

u. Click **Finish** to complete resource application creation.

v. Click **Activate** to activate your client application. The resource server representing the resource is now active.

(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 Infrastructure 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 **Next**.

11. In the **Client configuration** box, select **Configure this application as a client now**.
12. For authorization code, select **Refresh token** and **Authorization code** in the **Allowed grant types** section.
13. 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 Generation 2 to Oracle Integration Generation 2.
> - 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 Generation 2 instances	Yes.	`https://`*`OIC_instance_URL`*`.region.ocp.oraclecloud.com/icsapis/agent/oauth/callback`
Created on instances upgraded from Oracle Integration Generation 2 Generation 2 to Oracle Integration Generation 2	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.

14. Under **Resources**, 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.
15. Search for the Oracle Fusion Applications resource application created in Set Up Trust Between Oracle Fusion Applications and an Identity Domain.
16. Select the resource and click **>**.
17. Select the scope, then click **Add**.
18. Click **Next** without making changes on the Resource and Web Tier Policy pages.
19. On the Authorization page, click **Finish**.
The Application Added dialog 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. Activate the application.

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.

Set Up the OAuth Authorization Code Credentials Security Policy with Oracle Identity Cloud Service

You must set up trust between Oracle Fusion Applications and Oracle Identity Cloud Service and create a client application for Oracle Integration if you want to use the OAuth Authorization

Code Credentials security policy. Once these tasks are completed, you can successfully configure a connection on the Connections page.

> **Note:**
>
> The following instructions apply *only* to cloud tenancies that still use Oracle Identity Cloud Service. Most cloud tenancies have been migrated to identity domains, which require different configuration instructions. See Set Up the OAuth Authorization Code Credentials Security Policy with the Oracle Fusion Applications Identity Domain or Set Up the OAuth Authorization Code Credentials Security Policy with a Non-Oracle Fusion Applications Identity Domain. If you are unsure of your environment, check with your administrator.

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

Set Up Trust Between Oracle Fusion Applications and Oracle Identity Cloud Service

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

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

      See All REST Endpoints in *REST API for Oracle Identity Cloud Service*.
   b. Copy the endpoint.
   c. Get the Oracle Identity Cloud Service URL from the Oracle Cloud Infrastructure 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://IDCS_URL.identity.oraclecloud.com/admin/v1/SigningCert/jwk
      ```

   e. Perform a GET call to retrieve the payloads of the certificates. There are two sections in the payload:
      - Oracle Identity Cloud Service certificate
      - Certificate authority (CA) certificate

      Examples of the type of response you receive are provided. See Retrieve the Tenant's Signing Certificate in JWK Format.
   f. Copy both certificate 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_certificate
      ```

```
. . .
. . .
-----END CERTIFICATE-----
```

You can validate the certificate. For example:

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

2. Upload the certificates to the Oracle Fusion Applications Security Console.
   a. Log in to Oracle Fusion Applications as a user with the IT Security Manager role.
   b. In the navigation pane, select **Tools**, then **Security Console**.
   c. Select **API authentication** in the left navigation pane.
   d. Click **Create Oracle API Authentication Provider**, then click **Edit** in the upper right.
   e. In the **Trusted Issuer** field, enter:

      ```
      https://identity.oraclecloud.com
      ```

   f. In the **Token Types** section, select **JWT**.
   g. Click **Save and Close**.
   h. Click **Inbound API Authentication Public Certificates**, then click **Add New Certificate**.
   i. Enter a name in the **Certificate Alias** field (for example, `MY_IDCS_CERT`).
   j. In the **Import Public Certificate** field, click **Choose File** to upload the first certificate file, then click **Save**.
   k. Repeat these steps to upload the second certificate file.
3. Create an Oracle Identity Cloud Service resource application to represent the Oracle Fusion Applications resource.
   a. Log in to Oracle Identity Cloud Service as the Oracle Identity Cloud Service administrator.
   b. In the left navigation pane, click **Applications**, then click **Add**.
   c. Click **Confidential Application**.
   d. On the Details page, provide a name (for example, `FA Resource`), and click **Next**.
   e. On the Client page, click **Next** without making changes.
   f. On the Resources page, click **Configure this application as a resource server now**.
   g. Optionally update the value in the **Access Token Expiration** field.
   h. Select **Is Refresh Token Allowed**.
   i. 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
      ```

   j. In the **Scopes** section, click **Add**.
      The Add Scope dialog appears.
   k. In the **Scope** field, enter `/`.

l. In the **Display Name** field, enter a display name.

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

n. Select **Requires Consent**.

o. Click **Add**.

p. Click **Next** to move to the next page in the wizard.

q. On the Web Tier Policy page, click **Next** without making any changes.

r. On the Authorization page, click **Next** without making any changes.

s. Click **Finish** to complete resource application creation.

t. Click **Activate** to activate your client application. The resource server representing the resource is now active.

(Optional) Create a Local User

> **Note:**
>
> The following step is only required if the Oracle Fusion Applications user is *not* federated with Oracle Identity Cloud Service or whichever identity provider you are using.

1. Create an Oracle Identity Cloud Service 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 Oracle Identity Cloud Service that is protecting Oracle Integration.	No. You do not need to create the local Oracle Identity Cloud Service Oracle Fusion Applications user. This is because Oracle Identity Cloud Service already has Oracle Fusion Applications users in its repository.
You do *not* have federation between Oracle Fusion Applications and the Oracle Identity Cloud Service that is protecting Oracle Integration.	Yes. You must create the local Oracle Identity Cloud Service Oracle Fusion Applications user that you plan to use with the OAuth setup in Oracle Integration.

The Oracle Identity Cloud Service administrator must create a nonfederated local username in Oracle Identity Cloud Service 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 Oracle Identity Cloud Service and have a local user password.

Create the Confidential Client Application for Oracle Integration

1. Sign in as the Oracle Identity Cloud Service administrator to the Oracle Cloud Infrastructure Console. This administrator must have Oracle Identity Cloud Service instance access.
2. In the left navigation pane, select **Applications**, then click **Add** to add a client application.
3. Select **Confidential Application**.

The Add Confidential Application wizard is displayed.

4. On the Details page, enter an application name, and click **Next**.
5. On the Client page, click **Configure this application as a client now**.
6. In the **Authorization** section, select **Refresh Token** and **Authorization Code**.
7. In the **Redirect URL** field, enter your Oracle Integration instance URL and port. For example:

   ```
   https://OIC_URL:443/icsapis/agent/oauth/callback
   ```

   For the OAuth authorization code to work, the redirect URI must be set properly.
8. Under **Resources**, click **Add Scope** to add appropriate scopes.
   If the Oracle Fusion Applications instance is federated with the Oracle Identity Cloud Service instance, the Oracle Integration cloud service application is listed among the resources for selection. This enables the client application to access Oracle Integration.
9. Search for the Oracle Fusion Applications resource application created in Set Up Trust Between Oracle Fusion Applications and Oracle Identity Cloud Service.
10. Select the resource and click **>**.
11. Select the scope, then click **Add**.
12. Click **Next** without making changes on the Resource and Web Tier Policy pages.
13. On the Authorization page, click **Finish**.
    The Application Added dialog shows the client ID and client secret values.
14. 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...
Oracle Identity Cloud Service 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.
Oracle Identity Cloud Services 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 Oracle Identity Cloud Service application using the local Oracle Fusion Applications user account created earlier.

15. Activate the application.

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.

Enable Echo Suppression Filters

During bidirectional object synchronization, echoes are generated. For example, an event triggered in the Oracle CX Sales and B2B Service is synchronized through Oracle Integration to Oracle Service Cloud (with the Oracle RightNow Adapter), which then triggers an event in Oracle Service Cloud and then back to the Oracle CX Sales and B2B Service, and so on. Oracle Integration-based integrations use echo suppression to prevent unwanted update or create events (the echoes) from returning to the source application.

Filtering is used at the source application together with a `LastUpdatedBy` attribute in the record payload. In prebuilt integrations, Oracle Integration uses the Oracle CX Sales and B2B Service username created as part of the prebuilt integration setup to invoke every API of the application.

While the filtering is defined in Oracle Integration, it occurs in the event handler framework at activation/subscription time.

The Oracle CX Sales and B2B Service event handler framework evaluates the echo suppression expression after the internal event has been enriched to prevent the echo payload push to Oracle Integration.

- Echo suppression expression for account create/update flows is as follows:
  - `<xpathExpr xmlns:ns_0='http://xmlns.oracle.com/adf/svc/types/' xmlns:ns_1='http://xmlns.oracle.com/apps/crmCommon/salesParties/accountService/types/' xmlns:ns_5='http://xmlns.oracle.com/apps/crmCommon/salesParties/accountService/'>$eventPayload/ns_1:result/ns_0:Value/ns_5:LastUpdatedBy != 'CUSTOMER_OIC_INTEG_USER'</xpathExpr>`
- Echo suppression expression for contact create/update flows is as follows:
  - `<xpathExpr xmlns:ns_0='http://xmlns.oracle.com/adf/svc/types/' xmlns:ns_5='http://xmlns.oracle.com/apps/crmCommon/salesParties/contactService/' xmlns:ns_1='http://xmlns.oracle.com/apps/crmCommon/salesParties/contactService/types/'>$eventPayload/ns_1:result/ns_0:Value/ns_5:LastUpdatedBy != 'CUSTOMER_OIC_INTEG_USER'</xpathExpr>`

Integrate Groovy Scripts

The Oracle CX Sales and B2B Service Adapter inbound endpoint WSDL interface supports the use of concrete values for custom business objects in the WSDL in place of `xsd:anyType`

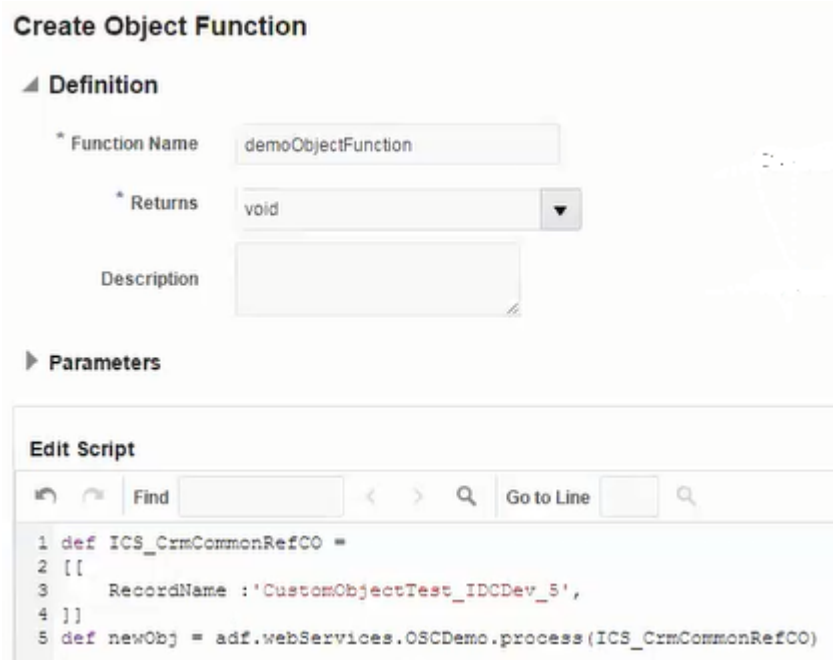
parameters. This feature enables you to use the groovy script editor to create scripts to invoke integrations. The groovy script editor is unable to recognize `xsd:anyType` parameters.

Perform the following high level steps from the CRM Fusion Home page (that is, the Fuse user interface) to integrate the groovy script.

> **Note:**
>
> If you have existing integrations with inbound endpoints (with custom business objects having `xsd:anyType`) generated using an Oracle Integration version prior to 16.3.3 and want to use groovy scripts, the integration must be re-edited and all artifacts must be generated again.

1. Create a sandbox environment in the Oracle CX Sales and B2B Service.
   a. In the upper right corner, select **Settings** > **Manage Sandboxes**.
   b. Create a sandbox.
   c. Select **Set as Active** to activate the sandbox.
2. On the desktop, click **Tools** > **Application Composer**.
3. Register the Oracle Integration web service.
   a. Click **Web Services**.
   b. Click the icon to create a new web service connection.
   c. Enter the name and WSDL URL, then click **Read WSDL**.
   d. Complete the other fields on the page, then save and close the page.
4. Create the object function to invoke the process method of the web service.
   a. In the left navigation pane, click **Object Workflows**.
   b. Expand **Objects** > **Custom Objects** > **CommonObject** > ***common_object_name***, then select **Server Scripts**.
   c. From the **Actions** dropdown list, select **Add**.
   d. Enter a function name and select a return value (for example, **void**).
   e. On the right side of the page, click **Show/Hide Function Palette**.
   f. Click **Web Services**.
   g. With the process function selected, click **Insert**.

      The Edit Script field is partially populated with the code for invoking the process method of the web service.
   h. Complete the script to create the object function. For example:



i. Click **Validate**.

j. Save and close.

5. Create the action to execute the function and expose the action as a button in the user interface.

   a. In the left navigation pane, click **Actions and Links**.

   b. From the **Actions** dropdown list, select **Create**.

   c. Enter a display label and name.

   d. From the **Method Name** dropdown list, select the method to use.

   e. On the right side of the page, select **Script** from the **Source** dropdown list.

   f. In the left navigation pane, click **Pages**.

   g. Click **Desktop Pages**.

   h. Click **Edit Creation Page**.

   i. In the **Configure Detail Form: Buttons and Actions** section, move the button you created to the **Selected Buttons** section.

   j. Save and close.

   k. Click the **Home** icon at the top of the page.

6. Execute the action by clicking the button.

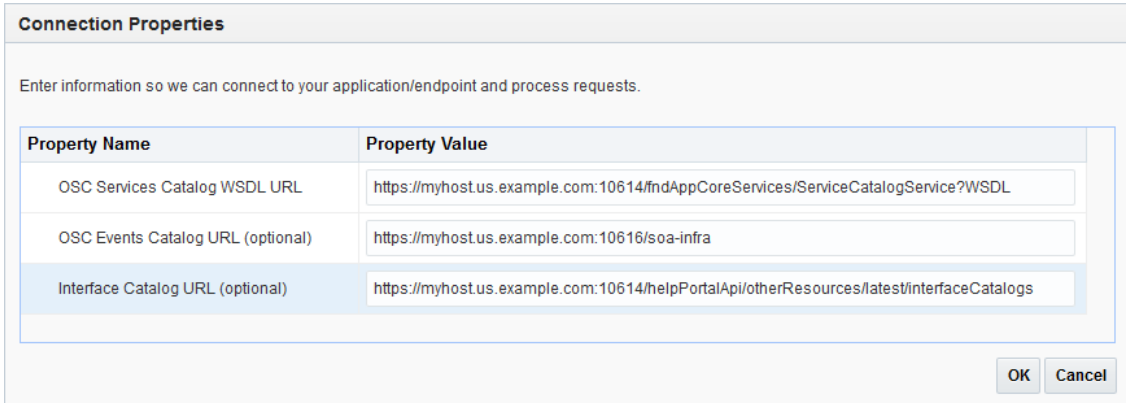
Obtain the Oracle CX Sales and B2B Service Service Catalog Service WSDL, Event Catalog URL, or Interface Catalog URL (For Connections Created Prior to 2/18/20)

The steps in this section are only required for existing connections created prior to the initial release of the simplified connections page on 2/18/20. For *existing* connections, you are prompted to specify a service catalog service WSDL (for accessing and configuring the

inbound and outbound adapter to use either business objects or business services) in the **OSC Services Catalog WSDL URL** field and an optional event catalog URL (for accessing and configuring the inbound adapter to use event subscriptions) in the **OSC Events Catalog URL** field or interface catalog URL (for accessing and configuring the outbound adapter to use Oracle Fusion Applications REST API resources) in the **Interface Catalog URL** field.

> **Note:**
>
> **For new connections created with the initial release of the simplified connections page on 2/18/20, the preconfiguration details described in this section are not required**. All WSDLs and URLs are automatically identified for you based on the Oracle CX Sales and B2B Service host name that you specify in the **CX Sales and B2B Service Host** field on the Connections page.

You specify the appropriate property values in the Oracle CX Sales and B2B Service Adapter Connection Properties dialog on the Connections page.



The following sections describe how to obtain the service catalog service WSDL, event catalog URL, and interface catalog URL:

- For Fusion Applications Releases 10 Through 12
- For Fusion Applications Releases 13 and Later

For Fusion Applications Releases 10 Through 12

Obtain the Oracle Fusion Applications Releases 10 through 12 service catalog service WSDLs, event catalog URLs, 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 Event Catalog URL
- 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 CX Sales and B2B Service connection must work with the Oracle CX Sales and B2B Service administrator to get the concrete WSDL URL for the service catalog service provisioned for the specific SaaS application.

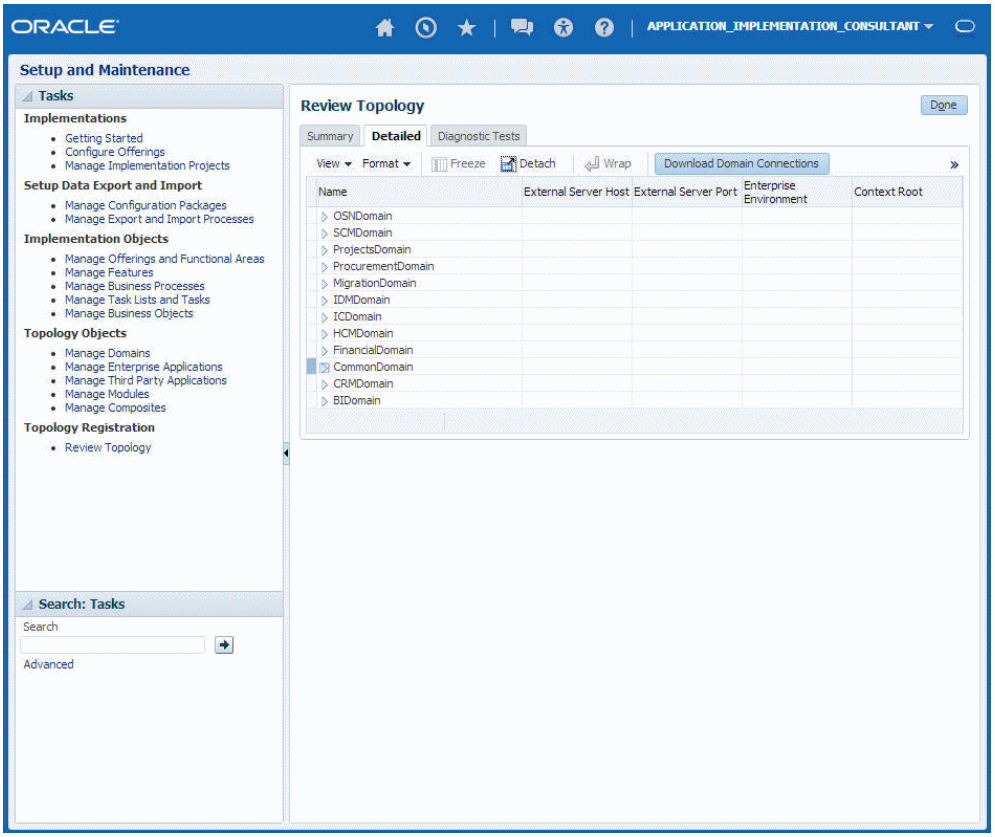
This section describes how to derive the external virtual host and port for a tokenized service catalog service WSDL. The topology information in the Topology Registration setup task contains the external virtual host and port for the domains and applications. The following instructions describe the 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.

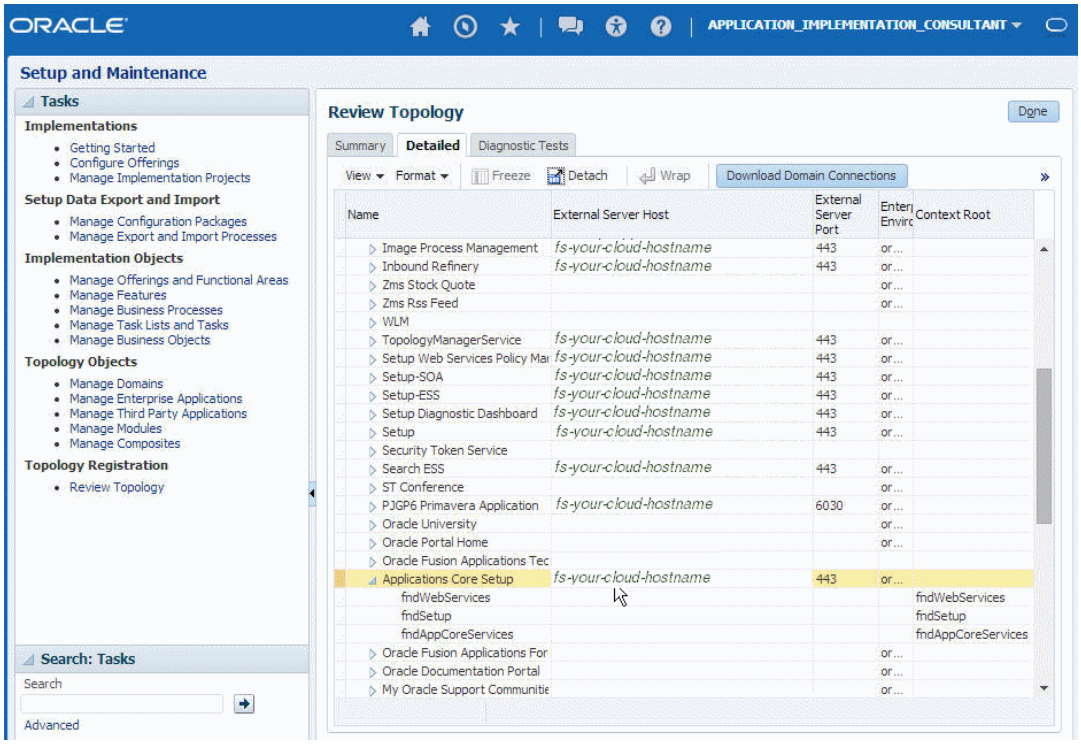
   The tab shows the list of domains configured in the cloud instance.



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 Event Catalog URL

You must know the customer relationship management (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
```

The event catalog URL `https://host:port/soa-infra` is a partial URL and must only be provided on the Connections page. Do not open this URL with a browser. If you do, you receive a `Page not found` error. The adapter does not access this URL directly. Instead, it automatically appends the required resource path to make the URL fully valid (when it needs to access the event catalog).

To check the public events defined in the catalog, enter the following complete URL in a browser:

```
https://host:port/soa-infra/PublicEvent/catalog
```

Obtain the Interface Catalog URL

The interface catalog URL is formatted as follows. Obtain the host name in the same way as you obtained the host for the service catalog service WSDL.

```
https://host:port_for_the_common_domain/helpPortalApi/otherResources/latest/
interfaceCatalogs
```

For Fusion Applications Releases 13 and Later

Obtain the Oracle Fusion Applications Release 13 service catalog service WSDL, event catalog URL, or interface catalog URL through the following methods.

- Obtain the Service Catalog Service WSDL
- Obtain the Event Catalog URL
- 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.fa.us6.oraclecloud.com/fscmService/ServiceCatalogService?WSDL
   ```

Obtain the Event Catalog URL

Starting in Release 13, you access all Fusion Applications URLs using a consolidated endpoint. You must switch to the new consolidated endpoint that conforms to the following naming pattern:

```
https://systemName.fa.dcsn.oraclecloud.com/...
```

You must switch to the consolidated endpoint immediately after upgrading to Release 13. Newly provisioned instances using Release 13 only have the consolidated endpoint available. In Release 13, multiple domains are consolidated. You must specify the domain in the URL. For example, if specifying the `fa` domain, the URL looks as follows:

1. Copy the following URL:

   ```
   https://acme.fa.us6.oraclecloud.com/
   ```

2. Append `soa-infra` to the end of the URL:

   ```
   https://acme.fa.us6.oraclecloud.com/soa-infra
   ```

Obtain the Interface Catalog URL

The interface catalog URL is formatted as follows. Obtain the host name in the same way as you obtained the host for the service catalog service WSDL.

```
https://host:port_for_the_common_domain/root_context/otherResources/latest/
interfaceCatalogs
```

Where *`root_context`* is one of the following root context values:

- ERP/SCM pillar:
  - `fscmRestApi`: for ERP/SCM REST services (includes PPM/Procurement).
  - `fscmService`: for ERP/SCM non-REST services, such as SOAP services.
- HCM pillar:
  - `hcmRestApi`: for HCM REST services.
  - `hcmService`: for HCM non-REST services, such as SOAP services.
- CRM pillar:
  - `crmRestApi`: for CRM REST services.
  - `crmServices`: for CRM non-REST services, such as SOAP services.

Create a Connection

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

To create a connection in Oracle Integration:

1. In the left navigation pane, click **Home** > **Integrations** > **Connections**.

2. Click **Create**.

> **Note:**
>
> You can also create a connection in the integration canvas of:
>
> - An orchestrated integration (See Define Inbound Triggers and Outbound Invokes.)
> - A basic routing integration (See Add a Trigger (Source) Connection.)

3. In the Create Connection — Select Adapter dialog, 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 and click

Search.
4. In the Create Connection dialog, enter the information that describes this connection.
   a. Enter a meaningful name to help others find your connection when they begin to create their own integrations. The name you enter is automatically added in capital letters to the **Identifier** field. If you modify the identifier name, don't include blank spaces (for example, `SALES OPPORTUNITY`).
   b. Enter optional keywords (tags). You can search on the connection keywords on the Connections page.
   c. Select the role (direction) in which to use this connection (trigger, invoke, or both). Only the roles supported by the adapter are displayed for selection. 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, let's say 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.
   d. Enter an optional description of the connection.
5. Click **Create**.

   Your connection is created. You're now ready to configure the connection details, such as connection properties, security policies, connection login credentials, and (for certain connections) agent group.

Configure Connection Properties

Enter connection information so your application can process requests.

1. Go to the **Connection 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 **Oracle CX Sales and B2B Service Host** field is displayed. Enter the Oracle CX Sales and B2B Service host name. For example:

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

> **Note:**
>
> The Oracle CX Sales and B2B Service host name can easily be derived from the Oracle CX Sales and B2B Service 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 **URL** fields are displayed. Specify the URLs to use in this integration.
   - OSC Services Catalog WSDL URL
   - OSC Events Catalog URL (optional)
   - Interface Catalog URL (optional)

   For information about obtaining the URL, see Obtain the Oracle CX Sales and B2B Service Service Catalog Service WSDL, Event Catalog URL, or Interface Catalog URL (For Connections Created Prior to 2/18/20).

Configure Connection Security

Configure security for your Oracle CX Sales and B2B Service Adapter connection by selecting the security policy and security token.

1. Go to the **Security** section.
2. Select the security policy to use.

Element	Description
Username Password Token	You receive the username and password to enter when subscribing to Oracle CX Sales and B2B Service Adapter. • **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 Set Up the OAuth Authorization Code Credentials Security Policy with Oracle Identity Cloud Service. • **Client Secret**: Enter the client secret issued during OAuth client application creation, then enter it a second time to confirm. See Set Up the OAuth Authorization Code Credentials Security Policy with Oracle Identity Cloud Service. • **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. • **Access Token URI**: Enter the URI to use for the access token. A request must be sent to this URI for obtaining an access token. • **Scope**: Enter the scopes specified during OAuth client application creation: – The URL that corresponds to the federated Oracle Fusion Application instance. – `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 Set Up the OAuth Authorization Code Credentials Security Policy with Oracle Identity Cloud Service. • **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 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 Oracle Identity Cloud Service Oracle Integration and Oracle Fusion Applications users are different, log in to the respective instance when prompted.

Element	Description
	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. See Avoid Potential Errors When Testing Your Connection with a Nonfederated User Account.

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, verify URLs and credentials, and download the diagnostic logs for additional details. Continue to test until the connection is successful.
3. When complete, click **Save**.

Upload an SSL Certificate

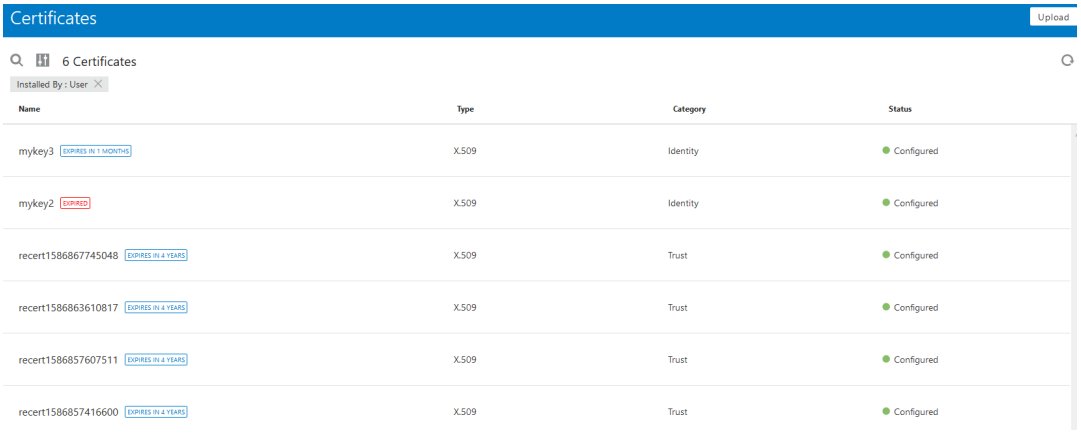
Certificates are used to validate outbound SSL connections. If you make an SSL connection in which the root certificate does not exist in Oracle Integration, an exception is thrown. In that case, you must upload the appropriate certificate. A certificate enables Oracle Integration to connect with external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

To upload an SSL certificate:

1. In the left navigation pane, click **Home** > **Settings** > **Certificates**.

All certificates currently uploaded to the trust store are displayed in the Certificates dialog. The

link enables you to filter by name, certificate expiration date, status, type, category, and installation method (user-installed or system-installed). Certificates installed by the system cannot be deleted.



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

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.
    c. 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.

To refresh integration metadata:

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

1. In the left navigation pane, click **Home** > **Integrations** > **Connections**.
2. Go to the row of the connection to refresh.
3. Select **Refresh Metadata** from the  menu.

   A message is displayed indicating that the refresh was successful.

   ```
   Metadata refresh for connection "connection_type" has been initiated
   successfully.
   ```

3

Add the Oracle CX Sales and B2B Service Adapter to an Integration

When you drag the Oracle CX Sales and B2B Service 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 CX Sales and B2B Service Adapter endpoint properties.

These topics describe the wizard pages that guide you through configuration of the Oracle CX Sales and B2B Service 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 CX Sales and B2B Service connection trigger request values for your integration. The values you specify start the integration.

Select the specific type to receive as a request from the Oracle CX Sales and B2B Service. Your ability to select either a business object or event subscription is based on the content of the WSDL file (for business objects) or event catalog URL (for event subscriptions) you specified during Oracle CX Sales and B2B Service Adapter configuration.

1. Select to receive a business object as a request from the Oracle CX Sales and B2B Service. This selection invokes the integration.
2. Select to receive an event subscription raised by the Oracle CX Sales and B2B Service application as a request from the Oracle CX Sales and B2B Service. This selection invokes the integration.

> **Note:**
>
> If business events are not displayed for selection, ensure that you configured your URL correctly on the Connections page. For connections created before the 18 February 2020 release of the simplified connections page, you must explicitly specify the Events Catalog URL. For connections created on or after the 18 February 2020 release of the simplified connections page, you must specify only the Oracle CX Sales and B2B Service host name. See Configure Connection Properties.

Element	Description
Configure a Request	Select the request type appropriate to your integration. The fields that are displayed below are based on the request type that you select. • **With Business Objects**: Select to display a list of business objects. • **With Business Events**: Select to display a list of event subscriptions
Select a Business Object (is displayed if **With Business Objects** is selected)	Select the business object from the Oracle CX Sales and B2B Service application to receive as a request that starts the integration.
Business Event For Subscription (is displayed if **With Business Events** is selected)	Select the event subscription from the Oracle CX Sales and B2B Service application to which to subscribe. This event is received as a request that starts the integration. Only events that can be subscribed to are displayed Any custom business events that you created and published in the the Application Composer application are also available for selection. Custom events are identifiable by their description.

Element	Description

Filter Expr for *Business_Event_Name*
(is displayed if **With Business Events** is selected)
	Enter an event condition filter expression. A filter expression specifies that the contents (payload or headers) of a message be analyzed before any event subscription is sent. For example, you can apply a filter expression that specifies that an event subscription be sent only if the message includes a customer ID. When the expression logic is satisfied, the event is accepted for delivery to the integration.
	As another example, assume you have the following event payload and want to process records where `OwnerName` is provided.
	
	`...`
	`<Body xmlns="http://`
	`schemas.xmlsoap.org/soap/envelope/">`
	`<ns01:onEvent`
	`xmlns:ns01="http://`
	`xmlns.oracle.com/cloud/adapter/osc/`
	`UpdateCustomerFromAccou nt_REQUEST/`
	`types">`
	`<ns0:getAccountResponse`
	`xmlns:wsu="http://docs.oasis-`
	`open.org/wss/2004/01/oasis-200401-`
	`wss-wssecurity-utility-1.0.xsd"`
	`xmlns:wsa="http://www.w3.org/`
	`2005/08/addressing"`
	`xmlns:ns0="http://`
	`xmlns.oracle.com/cloud/adapter/osc/`
	`UpdateCustomerFromAccoun t_REQUEST/`
	`types"`
	`xmlns:env="http://`
	`schemas.xmlsoap.org/soap/envelope/">`
	`<ns0:result`
	`xmlns:ns4="http://`
	`xmlns.oracle.com/apps/crmCommon/`
	`salesParties/accountService/"`
	`xmlns:xsi="http://www.w3.org/`
	`2001/XMLSchema-instance"`
	`xmlns:ns3="http://`
	`xmlns.oracle.com/apps/crmCommon/`
	`notes/noteService"`
	`xmlns:ns="http://`
	`xmlns.oracle.com/apps/crmCommon/`
	`salesParties/commonService /"`
	`xmlns:ns0="http://`
	`xmlns.oracle.com/apps/crmCommon/`
	`salesParties/accountServi ce/types/"`
	`xsi:type="ns4:Account">`
	
	`<ns4:PartyId>100000003362469</`
	`ns4:PartyId>`
	
	`<ns4:PartyNumber>AIMA-16BBRP</`

Element	Description

	ns4:PartyNumber>
	  <ns4:SourceSystem>CPI</
	ns4:SourceSystem>
	
	<ns4:SourceSystemReferenceValue>AIMA-
	16BBRP</
	ns4:SourceSystemReferenceValue >
	  <ns4:OrganizationName>Bank
	of America</ns4:OrganizationName>
	  <ns4:UniqueNameSuffix
	xsi:nil="true"/>
	  <ns4:PartyUniqueName>Bank of
	America</ns4:PartyUniqueName>
	  <ns4:Type>ZCA_CUSTOMER</
	ns4:Type>
	
	<ns4:OwnerPartyId>100000000225011</
	ns4:OwnerPartyId>
	  <ns4:OwnerPartyNumber>2008</
	ns4:OwnerPartyNumber>
	
	<ns4:OwnerEmailAddress>jhays@cpicardg
	roup.com</ns4:OwnerEmailAddress>
	  **<ns4:OwnerName>Jim Hays</
	ns4:OwnerName>**
	  . . .
	  </ns0:result>
	  </ns0:getAccountResponse>
	  </ns01:onEvent>
	</Body>
	
	The incoming event payload is referenced with `$inputVariable`. The payload response from the enrichment service can be referenced with `$eventPayload`. Given the event payload, you reference `OwnerName` as follows:
	
	<xpathExpr xmlns:ns0
	='http://xmlns.oracle.com/cloud/
	adapter/osc/
	UpdateCustomerFromAccount_REQUEST/
	types
	' xmlns:ns1='http://xmlns.oracle.com/
	apps/crmCommon/salesParties/
	accountService/'>
	$inputVariable/ns0:result/
	ns1:OwnerName = 'Jim Hays'</
	xpathExpr>

Filter by object name or **Filter By Event Name**	Type the initial letters of the name to filter the display of business objects or event subscriptions.

Trigger Response Page

Enter the Oracle CX Sales and B2B Service trigger response values for your integration.

You can configure the operation and business object that comprise the response type for the Oracle CX Sales and B2B Service.

- Immediate (synchronous) response: A response business object is immediately returned as output. You select **Immediate** as the response type on the Response page and select the business object as part of the response to the client. (See Response Type — Immediate (Synchronous) Response is Required for instructions.)
- Delayed (asynchronous) response: A callback service to which to route the callback is exposed. You select **Delayed** as the response type on the Response page and select the operation and business object that comprise a successful callback response, a failed callback response, or both. (See Response Type — Delayed (Asynchronous) Response is Required for instructions.)
- No response is required: You select **None** on the Response page because a response is not required. (See None — No Response is Required for instructions.)

The Response page looks as follows:

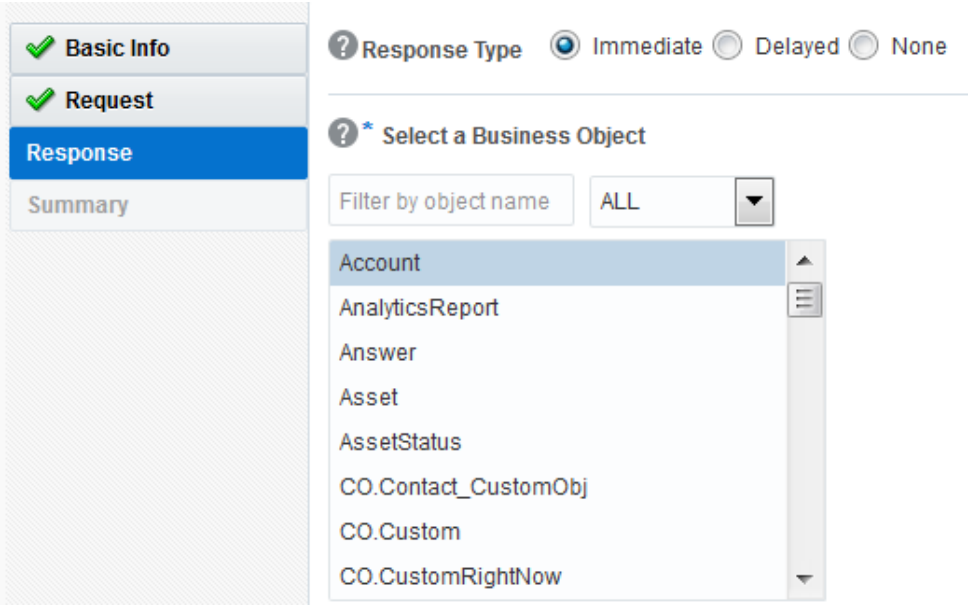


Table 3-1 Response Type — Immediate (Synchronous) Response is Required

Element	Description
Response Type	Select **Immediate** for the Oracle CX Sales and B2B Service application to wait until a response is received from the integration. This is also known as the request and response message exchange pattern. This is the default selection.
Filter by object name	Type the initial letters to filter the display of business objects.

Table 3-1 (Cont.) Response Type — Immediate (Synchronous) Response is Required

Element	Description
Select a Business Object	Select the business object to receive from the Oracle CX Sales and B2B Service application as a response. You can filter the display of business objects by typing the initial letters of business objects in the **Filter by object name** field. A description of the selected business object is displayed below this list.

Table 3-2 Response Type — Delayed (Asynchronous) Response is Required

Element	Description
Response Type	Select **Delayed** to configure a successful callback response, a failed callback response, or both. This enables you to configure the operation and business objects that you want the Oracle CX Sales and B2B Service application to process as part of a successful callback response, a failed callback response, or both.
Successful Response/Failed Response	Select the type of callback to configure. After configuring one type of callback (for example, successful), you can configure the other type (for example, failed). • **Successful Response**: Select to configure the operation and business objects that you want the Oracle CX Sales and B2B Service application to process as part of a successful callback response sent by the integration. • **Failed Response**: Select to configure the operation and business objects that you want the Oracle CX Sales and B2B Service application to process as part of an error callback response sent by the integration.
Select the operation to perform on the business object	Select the operation to perform on the business object.
Life Cycle	Displays the current state of the selected business document. Active indicates the business document is available for use. Deprecated indicates the business document is nearing the end of use and must be used with caution.
Description	Displays a description of the selected business object or service.

The following table describes the fields available if no response is required.

Table 3-3 None — No Response is Required

Element	Description
Response Type	Select **None**.
Select a Business Object	If you select **None**, this section is hidden.

Invoke Operations Page

Enter the Oracle CX Sales and B2B Service invoke operation values for your integration.

Select the type to browse:

- Business Objects
- Services
- Business (REST) Resource

Business Objects

Element	Description
Browse by	**Business Objects**: Select to browse by business object.
Select a Business Object	Select to filter the display of business objects, then select the business object to use. • **All** • **Active** • **Custom** • **Deprecated** • **Standard**
Filter by business object	Type the initial letters to filter the display of business objects.
Select the Operation to Perform on the Business Object	Select the operation to perform on the selected business object. The operations available for selection are based on the business object you selected.
Life Cycle	Displays the state of the selected business object. Deprecated indicates the business document is nearing the end of use and must be used with caution.
Description	Displays a description of the selected business object.

Services

Element	Description
Browse by	**Services**: Select to browse by service. There is a one-to-one correspondence between the business object and service. The service acts on the business document.
Select a Service	Select to filter the display of services, then select the service to use. • **All** • **Active** • **Deprecated** • **Standard**
Filter by service name	Type the initial letters to filter the display of services.

Element	Description
Select the Operation to Perform on the Business Service	Select the operation to perform on the selected service. The operations available for selection are based on the service you selected.
Life Cycle	Displays the state of the selected service. Deprecated indicates the business document is nearing the end of use and must be used with caution.
Description	Displays a description of the selected business service.

Business (REST) Resource

Element	Description
Browse by	**Business (REST) Resource**: Select to browse a list of available Oracle Fusion Applications REST API resources. This option is only available if you specified an interface catalog URL in the **Interface Catalog URL** field on the Connection Properties page when configuring the Oracle CX Sales and B2B Service Adapter. **Note**: The Oracle CX Sales and B2B Service Adapter currently pulls all resources exposed by the interface catalog and displays them for selection. For information about the public resources that are supported, see the Resource Types section of REST API for Oracle Sales Cloud Release 13.
Select a Service Application	Select the service application to see the business resources defined in the application.
Select a Business Resource	Select to filter the display of business resources, then select the business resource to use. • **All** • **Standard**
Filter by business resource	Type the initial letters to filter the display of business resources.
Select the Operation to Perform on the Selected Resource	Select the operation to perform on the selected business (REST) resource. The operations available for selection are based on the business resource you selected. You can also click **Browse and configure a child resource** to select the corresponding child business resources of that parent to use. **Note**: Existing integrations created prior to the introduction of the ability to select a child business resource can be edited to select parent business resources and their corresponding child business resources. **Note**: If you select **get**, only the following query parameters are supported: • **expand** • **fields** • **onlyData**

Element	Description
Browse and configure a child resource	Click to access a page to select the following: • The child business resources of the selected parent business resource • The operation to perform on the child 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**.

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.

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 **Back**. To cancel your configuration details, click **Cancel**.

4

Implement Common Patterns Using the Oracle CX Sales and B2B Service Adapter

You can use the Oracle CX Sales and B2B Service Adapter to implement 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.

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 CX Sales and B2B Service 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.

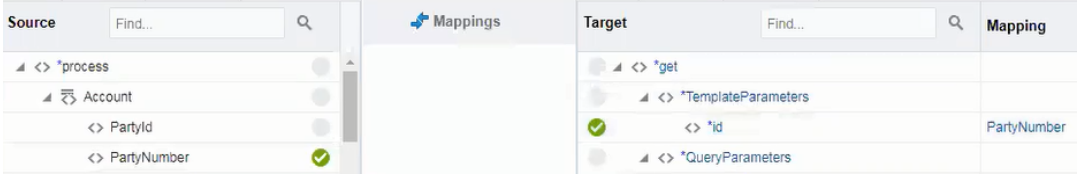
> **Note:**
>
> Note the following restrictions.
>
> - The request and response schema must be the same as provided during endpoint configuration.
> - Template parameters are not supported while mapping these properties.
> - The HTTP verb cannot be changed for the endpoint URL. For example, if the endpoint 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.

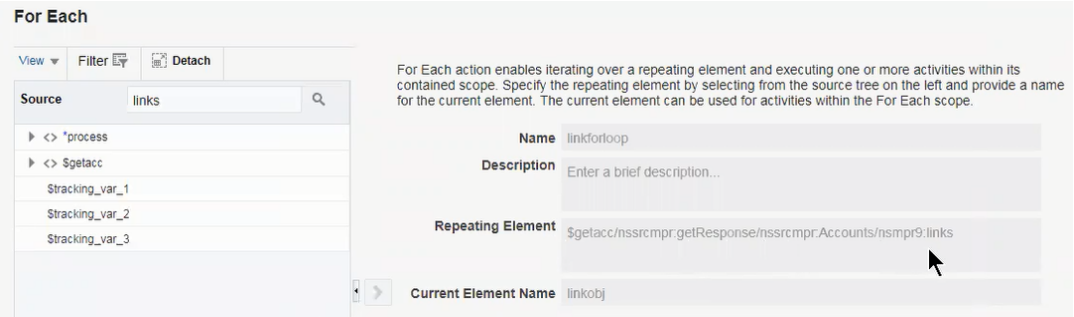
- An initial invoke is configured to get the Account object by using the REST API. The response of this REST API does not provide the child objects. Instead, there are HATEOS links to the child objects (that is, the Primary Address object).
- A second invoke uses the HATEOS links from the earlier response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

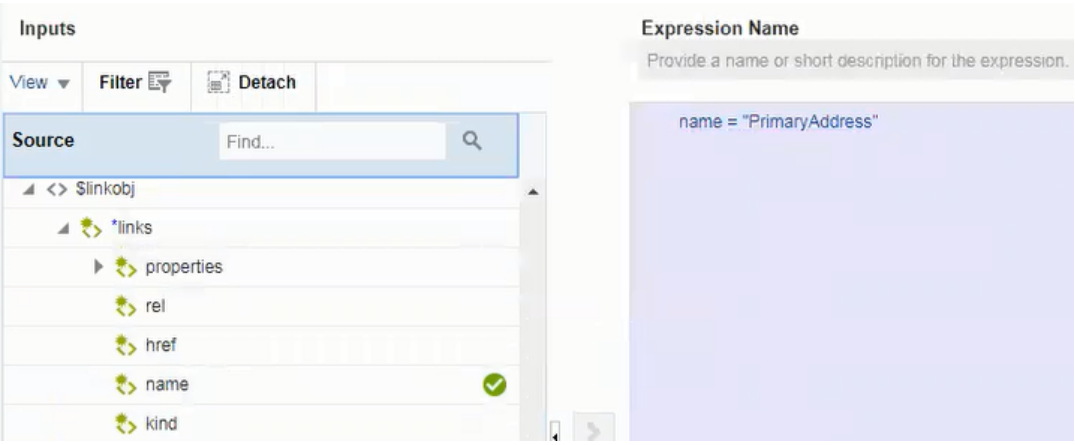
1. Create an orchestrated integration.
2. Drag an adapter into the integration canvas as an trigger connection (it can be any adapter).
3. Configure the adapter in the Adapter Endpoint Configuration Wizard.
4. Drag an initial Oracle CX Sales and B2B Service Adapter into the integration canvas as an invoke connection and configure it to use the **crmRestApp** service application and the **Account** object (business resource).
5. In the mapper between the trigger adapter connection and the Oracle CX Sales and B2B Service Adapter invoke connection, map source elements to target elements. For this example, a **PartyNumber** source element is passed to an **id** target element.



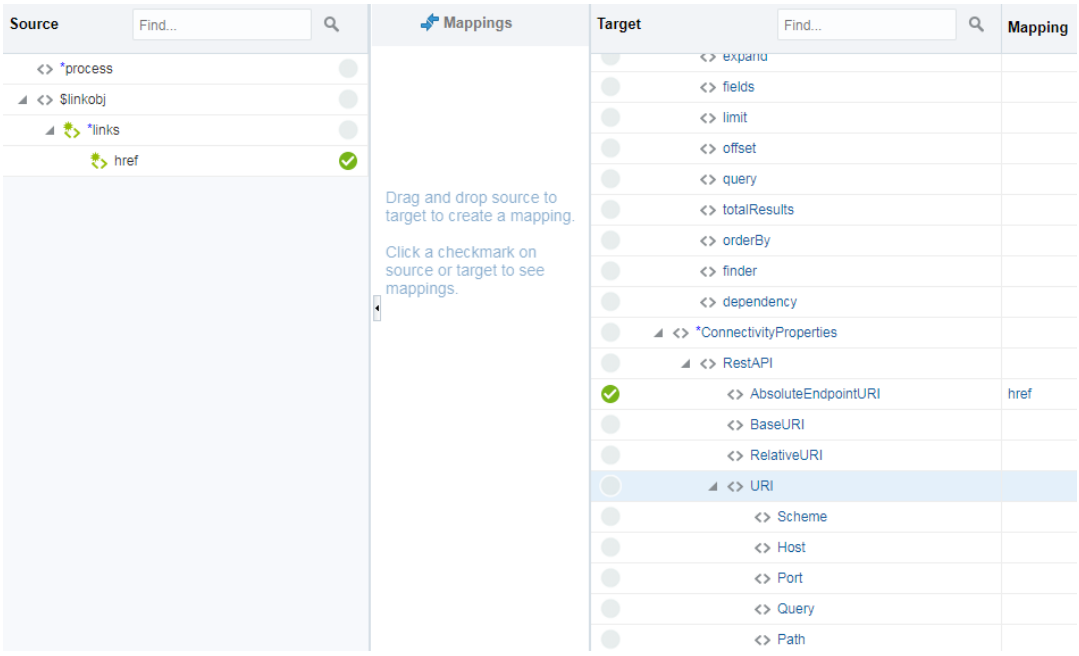
6. Add a for-each action to iterate between the HATEOS links. The value in the **Repeating Element** field is from the response object.



7. Add a switch action to get the HATEOS link corresponding to the **Primary Address** object.



8. Drag the Oracle CX Sales and B2B Service Adapter into the switch action as the second invoke connection and configure it to use the **crmRestApp** service application and the **Primary Address** object (business resource). 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.
9. In the mapper immediately before the second Oracle CX Sales and B2B Service Adapter invoke connection, expand **RestApi** under **ConnectivityProperties** in the target section.
10. 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 CX Sales and B2B Service Adapter.



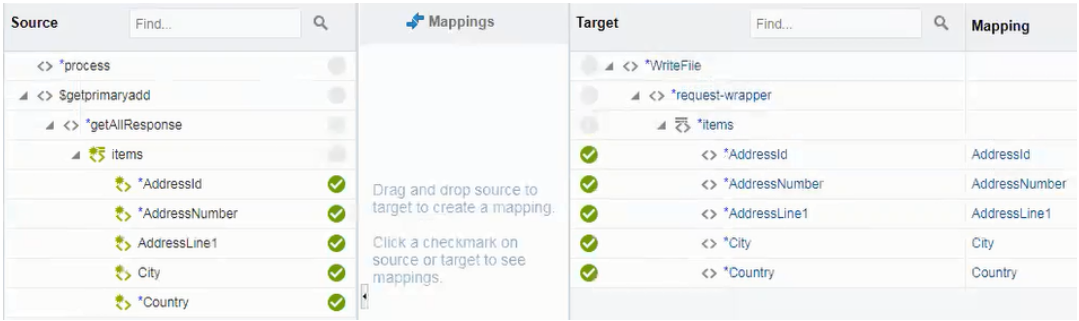
11. If necessary, map other nodes under **ConnectivityProperties**. The runtime values provided by these mappings dynamically configure the request.

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

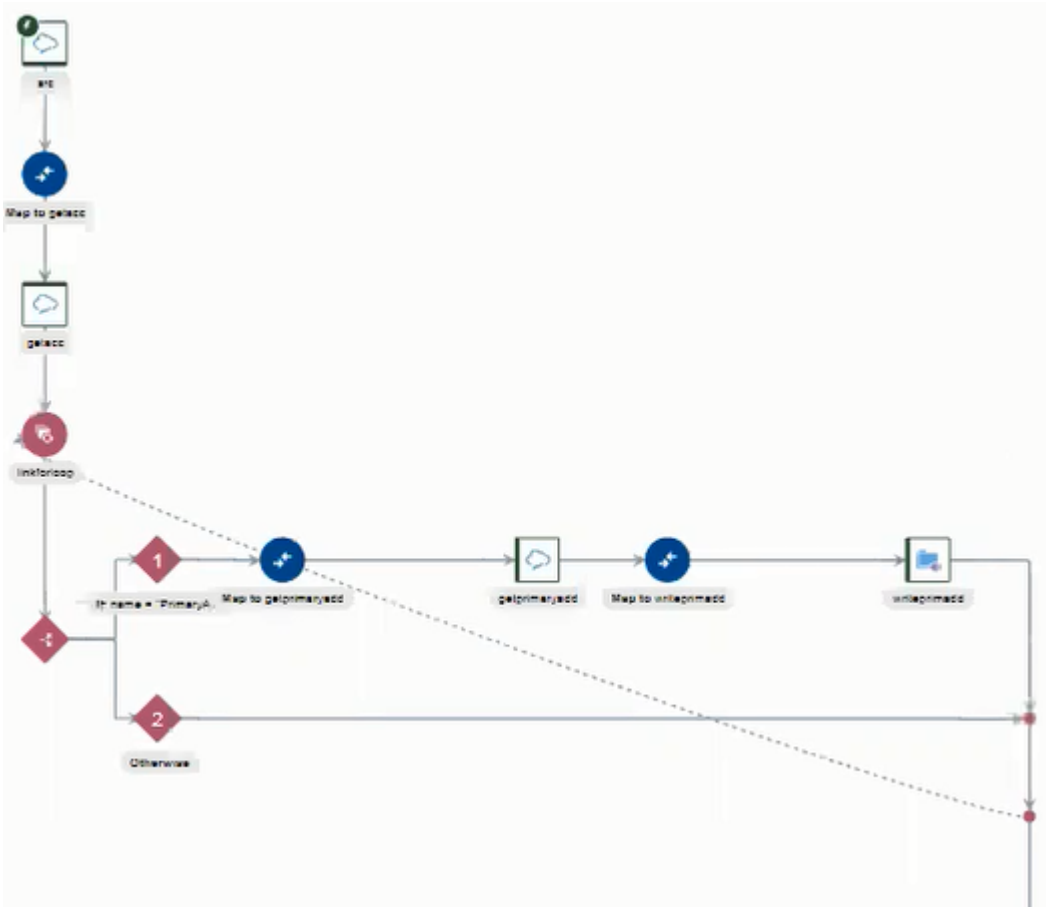
Element	Description
AbsoluteEndpointURI	Represents the absolute endpoint URL that the REST Adapter invokes. Empty values are ignored. To route the request to an endpoint URL determined at runtime, provide a mapping for this element. **AbsoluteEndpointURI** takes first precedence among other URL-related 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**.
Host	Provide a mapping to change only the **Host** portion of the endpoint URL

Element	Description
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.

12. Drag an FTP Adapter to the switch action for writing the **Primary Address** object response to a file on an FTP server.
13. In the mapper between the Oracle CX Sales and B2B Service Adapter and the FTP Adapter, map the **Primary Address** object details.



14. When complete, integration design looks as follows:



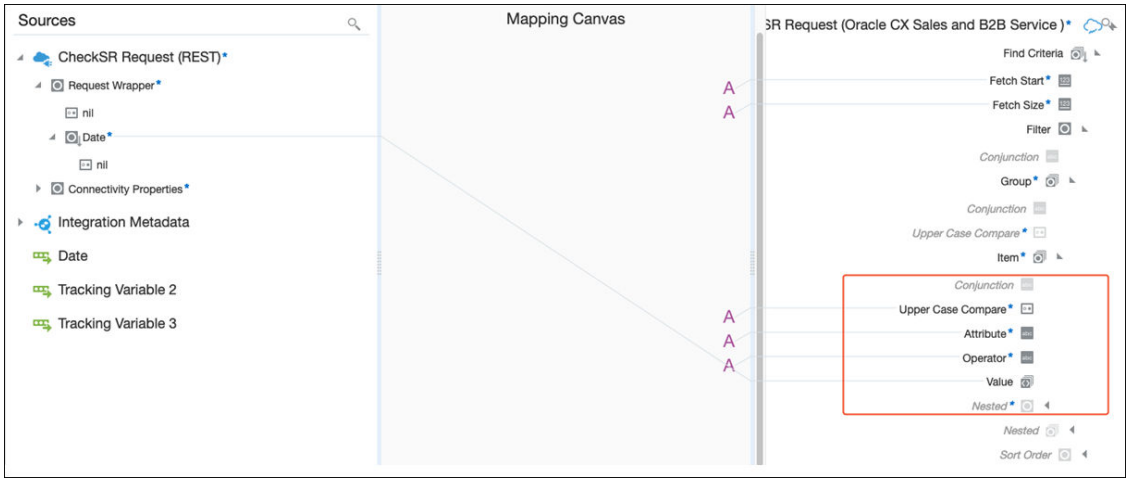
15. Activate and invoke the integration. The Oracle CX Sales and B2B Service Adapter invokes the endpoint URI determined at runtime.

Adaptive Search Queries

The Sales and Fusion Service REST API supports the Adaptive Search, which provides a faster and more organized search, and lets you save searches for future reference. You can manage the Adaptive Search using REST calls. The Adaptive Search supports multiple operators that adhere to cross-domain identity management (SCIM) standards.

See Adaptive Search Queries: Query Expression.

You can use the Adaptive Search to identify all the eligible operations in your use case. For example:



Add Multiple Users to an Exclusion List

When configuring the Oracle CX Sales and B2B Service Adapter as a trigger connection to consume Oracle Fusion Applications events, the Request page provides a field for specifying an event condition filter expression. This example describes how to create a filter expression that adds multiple users to an exclusion list through use of `or` and `contains` conditions.

When building this type of expression, ensure that you add parentheses around the content for both `contains` conditions:

```
(contains content) or (contains content)
```

For example:

```
<xpathExpr xmlns:ns_1='http://xmlns.oracle.com/apps/crmCommon/salesParties/
accountService/types/'
xmlns:ns_0='http://xmlns.oracle.com/adf/svc/types/' xmlns:ns_5='http://
xmlns.oracle.com/apps/crmCommon/salesParties/accountService/'>
not(contains($eventPayload/ns_1:result/ns_0:Value/
ns_5:LastUpdatedBy,'NO_EHF') or (contains($eventPayload/ns_1:result/
ns_0:Value/ns_5:LastUpdatedBy,'_MIGR')))
</xpathExpr>
```

Before an XPath expression is validated, spaces are removed. Without the parentheses, the expression is invalid. For example, `or contains` without the parenthesis becomes `orcontains`.

5

Troubleshoot the Oracle CX Sales and B2B Service Adapter

Review the following topics to learn about troubleshooting issues with the Oracle CX Sales and B2B Service Adapter.

Topics:

Additional integration troubleshooting information is provided. See Troubleshoot Oracle Integration in *Using Integrations in Oracle Integration Generation 2* and the Oracle Integration Troubleshooting page on the Oracle Help Center.

Deactivate an Integration with Business Events

If the integration to deactivate contains a business event subscription, a message is displayed asking if you want to delete the event subscription. If you select to delete the event subscription, the integration does not receive any events after it is reactivated.

If you do not want to delete the event subscription, the events in this integration are resent if the integration is activated within six hours.

Application Exception Not Routed to the Exception Handler Defined in the Integration

If the exception coming from Oracle Fusion Applications is not routed to the appropriate exception handler that you defined in the integration, this is because the Oracle Fusion Applications web service returned the fault code in the reply SOAP message as `env:Server`. This means it was returned as a generic SOAP exception and not as a business fault.

For this reason, the exception is not caught by the exception handler. When you receive the fault code under the SOAP message as `env:Server`, this is expected behavior and the message is not routed to the specific exception handler.