Oracle® Cloud

Using the Salesforce Commerce Cloud Adapter with Oracle Integration Generation 2

F32775-06
December 2022

Oracle Cloud Using the Salesforce Commerce Cloud Adapter with Oracle Integration Generation 2,

F32775-06

Primary Author: Oracle Corporation

Contents

4 Implement Common Patterns Using the Salesforce Commerce Cloud Adapter

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 Salesforce Commerce Cloud Adapter

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

Topics:

Salesforce Commerce Cloud Adapter Capabilities

The Salesforce Commerce Cloud Adapter enables you to create an integration with a Salesforce Commerce Cloud application. You can configure the Salesforce Commerce Cloud Adapter as an invoke connection in an integration in Oracle Integration.

The Salesforce Commerce Cloud Adapter provides the following benefits:

- Provides metadata caching support.
- Provides invoke (target) connection support for performing the following types of actions against the selected object and operation:
  - Create or Update (create or update a record in Salesforce Commerce Cloud)
  - Query (get records from Salesforce Commerce Cloud)
  - Delete (delete a record from Salesforce Commerce Cloud)
- Enables you to perform actions such as Create or Update, Query, and Delete on custom objects.
- Supports Salesforce Commerce Cloud APIs protected using OAuth 2.0 two-legged authentication.
- Supports Data and Shop APIs.

The Salesforce Commerce Cloud Adapter is one of many predefined adapters included with Oracle Integration. You can configure the Salesforce Commerce Cloud Adapter as an invoke connection in an integration in Oracle Integration.

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.

Workflow to Create and Add a Salesforce Commerce Cloud Adapter Connection to an Integration

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

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	Access Oracle Integration.	Go to `https://instance_URL/ic/home`
2	Create the adapter connections for the applications you want to integrate. The connections can be reused in multiple integrations and are typically created by the administrator.	Create a Salesforce Commerce Cloud Adapter Connection
3	Create the integration. When you do this, you add trigger (source) and invoke (target) connections to the integration.	Create Integrations in *Using Integrations in Oracle Integration Generation 2* and Add the Salesforce Commerce Cloud Adapter Connection to an Integration
4	Map data between the trigger connection data structure and the invoke connection data structure.	Map Data in *Using Integrations in Oracle Integration Generation 2*
5	(Optional) Create lookups that map the different values used by those applications to identify the same type of object (such as gender codes or country codes).	Manage Lookups in *Using Integrations in Oracle Integration Generation 2*
6	Activate the integration.	Activate Integrations in *Using Integrations in Oracle Integration Generation 2*
7	Monitor the integration on the dashboard.	Monitor Integrations in *Using Integrations in Oracle Integration Generation 2*
8	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*
9	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 a Salesforce Commerce Cloud Adapter Connection

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

Topics:

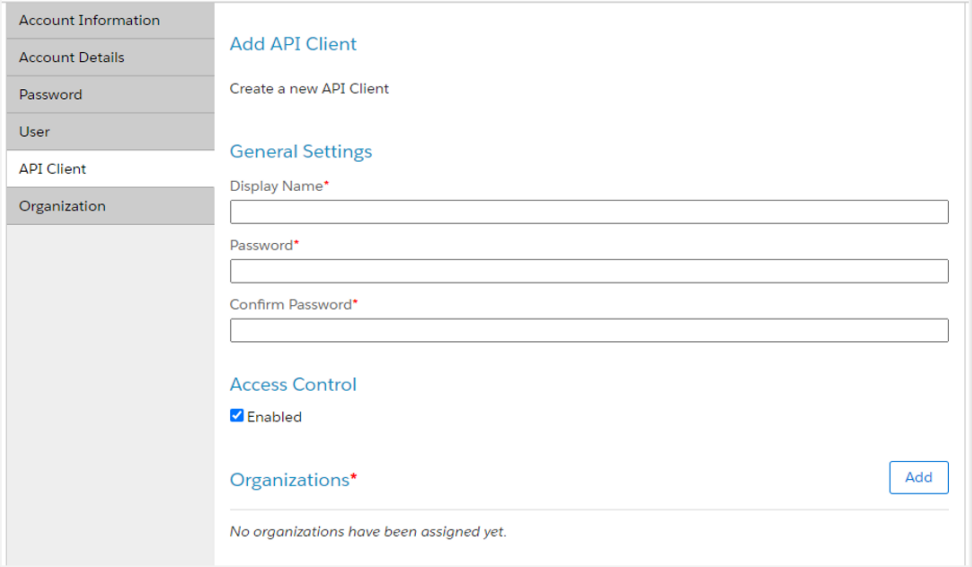
Prerequisites for Creating a Connection

You must satisfy the following prerequisites to create a connection with the Salesforce Commerce Cloud Adapter:

- Create an API Client
- Configure Open Commerce API Settings
- Find Your Current Salesforce Commerce Cloud API Version

Create an API Client

1. Log in to your Account Manager using the URL https://account.demandware.com.
2. Click **API Client**.
3. Click **Add API Client**. The Add API Client page is displayed.



4. Enter a display name and password, then confirm the password.
5. Copy the values for your API client password. You'll need to enter this value in the **Client Secret** field when you configure security for your Salesforce Commerce Cloud Adapter connection in Oracle Integration. See Configure Connection Security.
6. In the **Organizations** section, click **Add** to assign organizations to the API client. A list of organizations is displayed.
7. Select the organizations you want to assign to the API client and click **Add**.
8. (Optional) In the **Roles** section, click **Add** to assign roles to the API client. A list of roles is displayed.
9. Select the **Salesforce Commerce API** role and click **Add**.
10. After all the required details are entered, click **Save**.
11. When complete, click **API Client**. The details of your new API client are displayed in a list on the API Clients page and an API client ID is identified by the display name.

> **Note:**
>
> Use the client ID later when configuring security for your Salesforce Commerce Cloud Adapter connection. See Configure Connection Security.

Configure Open Commerce API Settings

> **Note:**
>
> - For the Shop API, the operations configured in the site context in the Open Commerce API (OCAPI) settings are displayed on the Operations page of the Adapter Endpoint Configuration Wizard. If there is no operation configured in the site context, the operations configured in the global context are displayed on the Operations page.
> - For the Data API, only the operations configured in the global context in the OCAPI settings are displayed on the Operations page of the Adapter Endpoint Configuration Wizard.

You must configure the OCAPI settings. These are mandatory settings required to manage OCAPI client permissions in order to invoke the APIs. You must provide permissions to the OCAPI client on the operations that you need to perform. Only the operations configured in the OCAPI settings are displayed on the Operations page of the Adapter Endpoint Configuration Wizard.

You can configure OCAPI settings. See Configuring OCAPI settings in Business Manager.

> **Note:**
>
> - The operations configured in the OCAPI settings may not be displayed instantly on the Operations page of the Adapter Endpoint Configuration Wizard because of the cache mechanism of Salesforce Commerce Cloud. You can invalidate the cache to see the configured operations without any delay.
> - If the operation is not configured in the OCAPI settings, OCAPI declines the client application request and returns an `HTTP 403 (Forbidden)` error status response code.

In addition, there are minimal access privileges (see the following table) that you must provide to the client in the OCAPI settings before you create a Salesforce Commerce Cloud Adapter connection. These APIs are used by the Salesforce Commerce Cloud Adapter to gather metadata information.

HTTP Method	API	Purpose
GET	/sites	Get a list of sites to populate the fields within site-specific fields.
GET	/locale_info/locales	Get a list of locale information to populate the fields within localized fields.
GET	/system_object_definitions/{object_type}/attribute_definitions	Get a list of attribute definitions for the **Product** object to make **variation value** keys in the mapper.

HTTP Method	API	Purpose
GET	/custom_object_definitions/{object_type}/attribute_definitions	Get a list of attribute definitions for the custom object.

Find Your Current Salesforce Commerce Cloud API Version

You can find your current Salesforce Commerce Cloud API version by accessing your own API, the meta API. The meta API provides information about the available APIs.

1. Open a browser and enter the URL in the following format:

   `https://my_instance/s/-/dw/meta/rest/data`

   A list of API versions released by the Salesforce Commerce Cloud instance is displayed.
2. In the displayed list, the **name** field shows the API version of the Salesforce Commerce Cloud instance and the **status** field shows the status of the API version such as obsolete, deprecated, or current.

   > **Note:**
   >
   > The Salesforce Commerce Cloud Adapter supports current and deprecated API versions.

3. Scroll down to the bottom and find the API version with status `current`. For example:

   `"name":"20.10","status":"current"`

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.
2. In the **Host** field, enter the host URL in the following format.

    `https://my_instance`

3. In the **API Version** field, enter the API version to use. See Find Your Current Salesforce Commerce Cloud API Version.
4. Click **OK**.

Configure Connection Security

Configure security for your Salesforce Commerce Cloud Adapter connection.

You can select the Client Credentials security policy to execute the Data API. Or you can select the Business Manager user grant security policy to execute both the Shop and Data APIs. The Shop API is not supported when you configure the Client Credentials security policy for your Salesforce Commerce Cloud Adapter connection.

> **Note:**
>
> The Alerts and Ocapi Configs objects of Data API are only supported when you configure the Business Manager user grant security policy for your Salesforce Commerce Cloud Adapter connection.

1. Go to the **Security** section.
2. From the **Security Policy** list, select the security policy.
   - **Business Manager user grant**: Select this security policy to execute both Shop and Data APIs.
   - **Client Credentials**: Select this security policy if you need to execute only the Data API. The Shop API is not supported when you configure the Client Credentials security policy for your Salesforce Commerce Cloud Adapter connection.
3. If you select **Business Manager user grant**.
   a. In the **Username** field, enter the username.
   b. In the **Password** field, enter the password.
   c. In the **Client Id** field, enter the client ID that you obtained after performing the steps in the prerequisites section. See Create an API Client.
   d. In the **Client Secret** field, enter the client secret that you obtained after performing the steps in the prerequisites section. See Create an API Client.
4. If you select **Client Credentials**.
   a. In the **Client Id** field, enter the client ID that you obtained after performing the steps in the prerequisites section. See Create an API Client.
   b. In the **Client Secret** field, enter the client secret that you obtained after performing the steps in the prerequisites section. See Create an API Client.
5. Click **OK**.

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 connection uses a Web Services Description Language (WSDL) file.

If Your Connection...	Then...
Doesn't use a WSDL	The test starts automatically and validates the inputs you provided for the connection.

If Your Connection...	Then...
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.

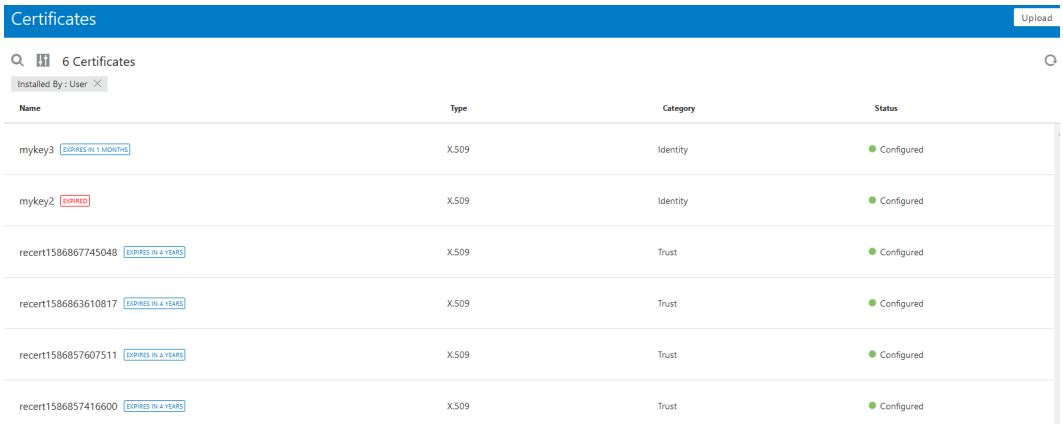
2. Wait for a message about the results of the connection test.
   - If the test was successful, then the connection is configured properly.
   - If the test failed, then edit the configuration details you entered. Check for typos, verify URLs and credentials, and download the diagnostic logs for additional 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 Salesforce Commerce Cloud Adapter Connection to an Integration

When you drag the Salesforce Commerce Cloud Adapter into the invoke area of an integration, the Adapter Endpoint Configuration Wizard is invoked. This wizard guides you through configuration of the Salesforce Commerce Cloud Adapter endpoint properties.

The following sections describe the wizard pages that guide you through configuration of the Salesforce Commerce Cloud Adapter as an invoke in an integration. The Salesforce Commerce Cloud Adapter cannot be used as a trigger 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.`

Element	Description
Select API type	Select the API type. • **Data API** • **Shop API** **Note**: • You must configure the Business Manager user grant security policy for your Salesforce Commerce Cloud Adapter connection to execute the Data and Shop APIs. • The Shop API is not supported when you configure the Client Credentials security policy for your Salesforce Commerce Cloud Adapter connection. See Configure Connection Security.
Select site (Only displayed if you selected the **Shop API** option.)	Select the site, such as Ref Arch, Site Genesis, Site Genesis Global, and so on.

Invoke Action Page

Select the type of action to perform:

Element	Description
Select action type	• **Create or Update**: Creates a new record or updates the existing record in the Salesforce Commerce Cloud application. • **Delete**: Deletes the record from the Salesforce Commerce Cloud application. • **Query**: Retrieves information from the Salesforce Commerce Cloud application corresponding to the selected object and operation.

Invoke Operations Page

Select the object and operation to perform in the Salesforce Commerce Cloud application.

Element	Description
Select object type	• **Standard objects**: Displays business objects delivered as part of the Salesforce Commerce Cloud application. • **Custom objects**: Provides an option to search custom objects you created and shows the custom objects accordingly.

The fields that appear are based on the object type you select. If you select **Standard objects**, the following options are displayed.

Element	Description
Select operation type (Displayed if the **Query** action is selected on the Action page)	• **Retrieve/GET**: Retrieves records from the Salesforce Commerce Cloud application. • **Query/Search**: Returns records from the Salesforce Commerce Cloud application.
Select object	Use the scrolling list to select an object within the selected action.
Filter by object name	Type the initial letters of the object name to filter the display of names in the list.
Select operation	Select an operation name, such as **Put User By ID**.
Filter by operation name	Type the initial letters of the operation name to filter the display of names in the list.
Click to Configure Query (Displayed if the **Query/Search** option is selected)	Click to configure the query against the Salesforce Commerce Cloud application.

The following options are displayed when you click the **Configure Query** button.

Element	Description
Enter your query	Enter a valid query statement. You can add binding parameters in a query to pass dynamic values at runtime. Add the binding parameters in the allowed positions (that is, `values` and `search_phrase`) in the statement. Add the ampersand symbol (`&`) to a variable value to make it a binding parameter (for example, `&cat`). These parameters are displayed in the request mapper for mapping.
Execute Query	Click to validate the query against the Salesforce Commerce Cloud application.
Binding parameters	Displays bind variables included in the query.
Batch size	Specifies the batch size for queries. The batch size determines the maximum number of records returned for each query. The default value is 200, the minimum allowed size is 2, and the maximum is 200.
Query result	Displays query results.

If you select **Custom objects**, the following options are displayed.

Element	Description
Select operation type (Displayed if the **Query** action is selected on the Action page)	**Retrieve/GET**: Retrieves records from the Salesforce Commerce Cloud application.
Enter custom object type	Enter a custom object type.

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**. Click **generate a sample cURL** to generate sample `cURL` syntax for the configuration options that you have selected during REST Adapter connection configuration, such as security policy, headers, parameters, and so on.

4

Implement Common Patterns Using the Salesforce Commerce Cloud Adapter

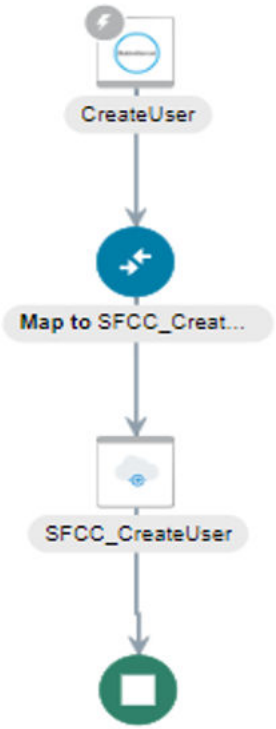
You can use the Salesforce Commerce Cloud Adapter to implement the following common patterns.

Topics:

Synchronize a Salesforce User with a User in the Salesforce Commerce Cloud Application

This use case provides an overview of how to synchronize a Salesforce user with a user in the Salesforce Commerce Cloud application.

1. Create an app-driven orchestrated integration.
2. Drag a Salesforce Adapter into the integration canvas.
3. Configure the workflow rule and outbound messages in the Salesforce application. The same WSDL must be uploaded in the integration.
4. Drag a Salesforce Commerce Cloud Adapter.
5. Configure the Salesforce Commerce Cloud endpoint:
    a. On the Basic info page, provide an endpoint name, and click **Next**.
    b. On the Action page, select **Create or Update**, and click **Next**.
    c. On the Operations page, select **Standard objects**, select users as an object, select **Put User By ID** as an operation, and click **Next**.
    d. On the Summary page, review your selections, then click **Done**.
6. In the mapper, map the **login** in the template parameter and the **First Name**, **Last Name**, and **Email** elements to the respective fields of Salesforce Commerce Cloud. The completed integration looks as follows.



7. Specify the tracking variable, save, and close the integration.
8. Activate the integration.
9. Create a user in Salesforce. The same user is now reflected in the Salesforce Commerce Cloud application.
10. To verify the created user, log in to your Salesforce Commerce Cloud account, go to **Administration** > **Organization** > **Users**, and click **Find**. The details of your new user are displayed in a list on the User List page.

Use the Pagination Concept in an Integration

When you must fetch a large number of results in Salesforce Commerce Cloud, you can use the Salesforce Commerce Cloud Adapter's pagination feature. Pagination helps you segregate the total number of results. You can specify the number of results per page according to your requirement while triggering an integration and receive sorted results.

This use case discusses how to use the pagination to sort results while fetching catalog results. Similarly, you can use pagination for the Stores object, Roles object, Products object, and so on. To perform this operation, you create an integration for the Salesforce Commerce Cloud Adapter in Oracle Integration using pagination with the orchestrator.

1. Create an app-driven orchestrated integration.
2. Drag the SOAP Adapter connection into the integration as a trigger connection.
3. Drag an assign action to the workspace next to the SOAP Adapter.
4. Provide a unique name for this action, and click **Create**.

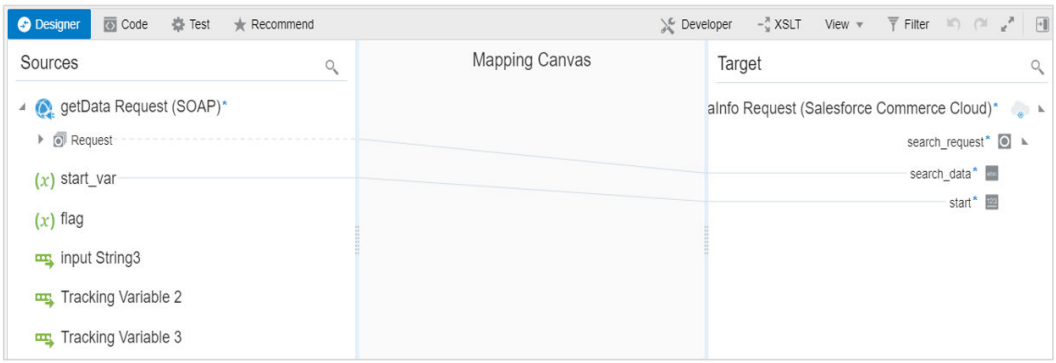
5. Assign the following two variables to your integration. Specify the values to variables as follows.

```
Start = 0.0
flag = "true"
```

6. Click **Validate**, and then click **Close**.
7. Drag a while action below the assign action.
8. Provide a unique name for this action, and click **Create**.
9. Set the condition for the specified variables as follows.

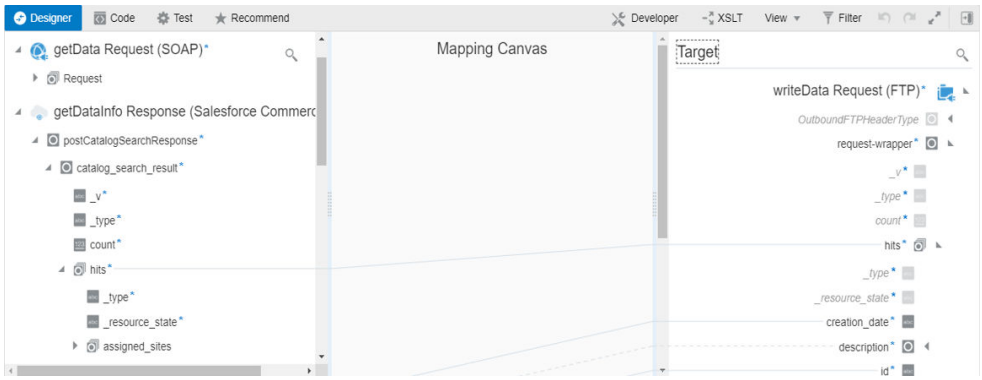
```
flag! = "false"
```

10. Click **Validate**, then click **Close**.
11. In the mapper, map the **start assign variable** to **start** in the Salesforce Commerce Cloud request mapper and the input from **SOAP** to **search_data**.



> **Note:**
>
> The **start** element is a numeric value required to search and access a specific record in the results. The **search_data** is a text field designed for the user to enter search queries into.

12. Drag a Salesforce Commerce Cloud Adapter inside a while action.
13. Specify the following details in the Adapter Endpoint Configuration Wizard.
    a. On the Basic Info page, provide a name.
    b. On the Actions page, select the **Query** action.
    c. On the Operations page, select **Query/Search** as an operation type, select **catalog_search** as an object, and select **Post Catalog Search** as an operation.
    d. Review your selections on the Summary page.
14. Drag an FTP adapter and configure it with `.json` data file.
15. Perform the required mapping.



16. Drag a switch action and specify the following IF condition:

    ```
    start<=total
    $getDataInfo/nsmpr8:postCatalogSearchResponse/nsmpr8:
    catalog_search_result/nsmpr9:next/nsmpr8:start<= $getDataInfo/
    nsmpr8:postCatalogSearchResponse/nsmpr8:catalog_search_result/
    nsmpr9:total
    ```

17. Drag an assign action to update the specified variables as follows:

    ```
    start_var = start
    flag = "true"
    ```

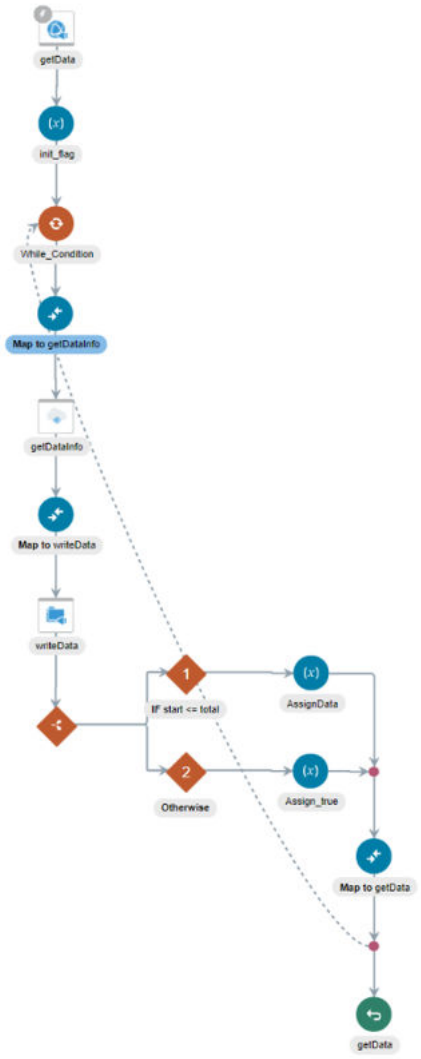
    The while loop runs next in Salesforce Commerce Cloud. Each page gets triggered and fetches the required data.

18. Drag an assign action in the Otherwise branch of the switch action to update the specified variables as follows:

    ```
    flag = "false"
    ```

19. When complete, save and activate the integration.

    The completed integration looks as follows.



20. After successful activation, you can submit the integration and monitor the runtime in Oracle Integration. The integration keeps running until it completes all cycles of fetching product results, respectively, to each page in the FTP location.

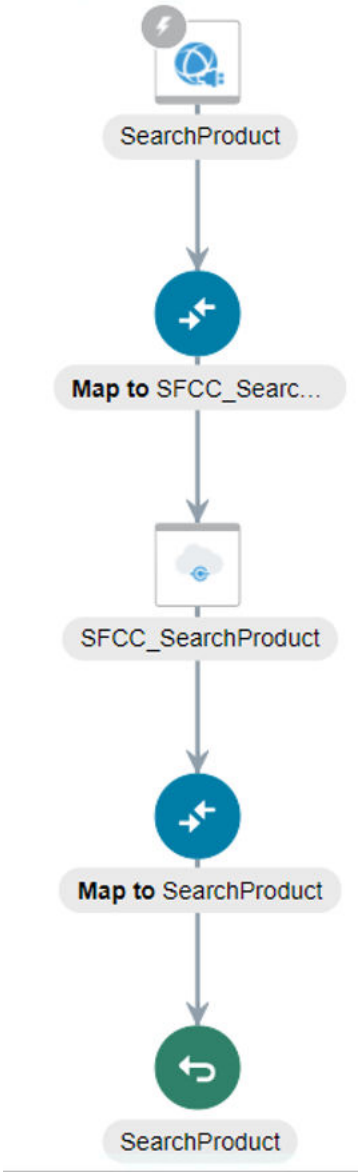
Fetch Records in Salesforce Commerce Cloud Using the Query Operation

When you need to fetch records in Salesforce Commerce Cloud, you can use the Salesforce Commerce Cloud Adapter's query/search feature. You can specify the query statement according to your requirement and fetch records.

This use case describes how to use the query/search operation to fetch the matching records from the Product object.

1. Create an app-driven orchestrated integration.
2. Drag a SOAP Adapter into the integration as a trigger connection.
3. Drag an assign action to the workspace next to the SOAP Adapter.

4. Provide a unique name for this action, and click **Create**.
5. Drag a Salesforce Commerce Cloud Adapter into the integration.
6. Configure the Salesforce Commerce Cloud endpoint:
    a. On the Basic info page, provide an endpoint name, and click **Next**.
    b. On the Action page, select **Query**, and click **Next**.
    c. On the Operations page, select **Query/Search** as an operation type, select **product_search** as an object, select **Post Product Search** as an operation, and click **Click to Configure Query**.
    d. Enter the query statement according to your requirement, and click **Execute Query**. The matching results are displayed in the **Query result** field.
    e. Review the search results, and click **Ok**.
    f. On the Summary page, review your selections, and click **Done**.
7. In the mapper, map the **search_data** and **start** parameters in the request mapper and query result elements to the respective fields of Salesforce Commerce Cloud in the response mapper. The completed integration looks as follows.



8. Specify the tracking variable, save, and close the integration.
9. Activate the integration.
10. After successful activation, you can submit the integration and monitor the runtime in Oracle Integration. The integration keeps running until it completes the search for fetching product results as the output response.