ServiceNow Adapter Guide

For SAP Integration Suite and SAP Cloud Integration

Version 1.0.6 – May 2026

THE BEST RUN SAP

Table of Contents

1. INTRODUCTION **3**
1.1 Coding Samples **3**
1.2 Internet Hyperlinks **3**
2. SERVICENOW INTEGRATION **3**
2.1 Introduction **3**
2.2 ServiceNow Adapter **3**
2.2.1 Features ***3***
2.3 Architecture Overview **4**
3. SUPPORTED OPERATIONS AND VERSIONS **5**
4. ADAPTER INSTALLATION **5**
5. ECLIPSE PLUG-IN INSTALLATION **5**
6. USING THE ECLIPSE PLUG-IN **5**
7. AUTHENTICATION **8**
7.1 Supported Authentication Mechanisms **8**
7.2 Creating User Credential in Security Material **8**
7.3 Creating Secure Parameter in Security Material **8**
7.4 Configuration of Authentication in the Adapter **8**
8. RECEIVER ADAPTER CONFIGURATION **9**
8.1 Connection Tab **9**
8.2 Processing Tab **10**
8.2.1 Create Object ***14***
8.2.2 Delete Object ***15***
8.2.3 Get Object by ID ***16***
8.2.4 Modify Object ***17***
8.2.5 Update Object ***18***
8.2.6 Query Objects ***19***
8.2.7 Query Objects (Advanced) ***24***
8.2.8 Attachment – Delete ***27***
8.2.9 Attachment – Query Metadata ***28***
8.2.10 Attachment – Retrieve Content ***28***
8.2.11 Attachment – Retrieve Metadata ***28***
8.2.12 Attachment – Upload Binary ***29***
9. SAMPLE SCENARIOS EXPLAINED **30**
10. HEADERS AND HTTP CODES **33**
10.1 Headers **33**
10.2 HTTP Response Codes **33**
11. SUPPORT AND TROUBLESHOOTING **34**
11.1 SSL Handshake Exception **34**
11.2 No Artifact Descriptor Found **35**
11.3 Invalid Table xxxx **35**
12. REFERENCES **35**
12.1 Create and OAuth Endpoint for the ServiceNow Adapter **35**

1. INTRODUCTION

This is the official guide for the ServiceNow Adapter for Cloud Integration. This guide covers all relevant information for integration developers to start working with the ServiceNow Adapter. Please read this guide carefully before using the ServiceNow Adapter.

1.1 Coding Samples

Any software coding and/or code lines/strings ("Code") included in this documentation are examples. These are not intended for use in a productive system environment. The Code is intended to better explain and visualize the syntax and phrasing rules of certain coding. We do not guarantee the correctness and completeness of the Code given herein.

1.2 Internet Hyperlinks

The documentation may contain hyperlinks to the Internet. These hyperlinks serve as hints on where to find related information. We do not warrant the availability and the correctness of this related information or the ability of this information to serve a particular purpose.

2. SERVICENOW INTEGRATION

2.1 Introduction

ServiceNow is an IT Service Desk and ticketing platform that processes and catalogs customer service requests. It can be used to raise requests that deal with incidents, changes, problems, etc. It can also be used for Asset Management, Change Management, Human Resources, etc.

2.2 ServiceNow Adapter

The ServiceNow receiver adapter enables the SAP Cloud Integration tenant to accelerate the implementation time and reduce the complexity of connecting to ServiceNow.

2.2.1 Features

From a high-level perspective, the following key features are provided by the ServiceNow Adapter:

- Support multiple versions: The Adapter lets the User integrate with ServiceNow using a preferred API version. At the moment of publishing this documentation, the Adapter supports ServiceNow API v1, v2, and Latest.
- Support multiple operations, such as:
    - Create Object
    - Delete Object
    - Get Object by ID
    - Modify Object
    - Query Objects
    - Query Object (Advanced)
    - Update Object
    - Attachment – Delete
    - Attachment – Query Metadata
    - Attachment – Retrieve Content
    - Attachment – Retrieve Metadata
    - Attachment – Upload Binary
- Support outbound headers.
- Support retrieval of different page results using Limits and Offsets.
- Support dynamic configuration with headers and properties.

- Dynamic configuration with headers and properties. It is possible to set up scenarios as dynamic as desired. Refer to dynamic values from the header and properties stored in the Exchange section of the SAP CPI tenant.
- Support for XML and JSON formats. The Adapter can handle both XML and JSON input and response messages.
- Support Basic and OAuth as authentication mechanisms.
- XSD generation. The Adapter is equipped with an Eclipse Plug-in which generates XSDs representing the tables in ServiceNow.

2.3 Architecture Overview

From a technical perspective, the ServiceNow Adapter can be used as a receiver adapter where the SAP Cloud Integration acts as the initiator of the calls. In case the calls toward ServiceNow need to be scheduled, use the Scheduler step within the integration flow.

Figure 2.1 shows how the ServiceNow Adapter can be used in a simple Request-Reply scenario. Various operations can be used in the Adapter. These are explained in Section 3. Figure 2.1 gives a high-level representation of how the Adapter works.

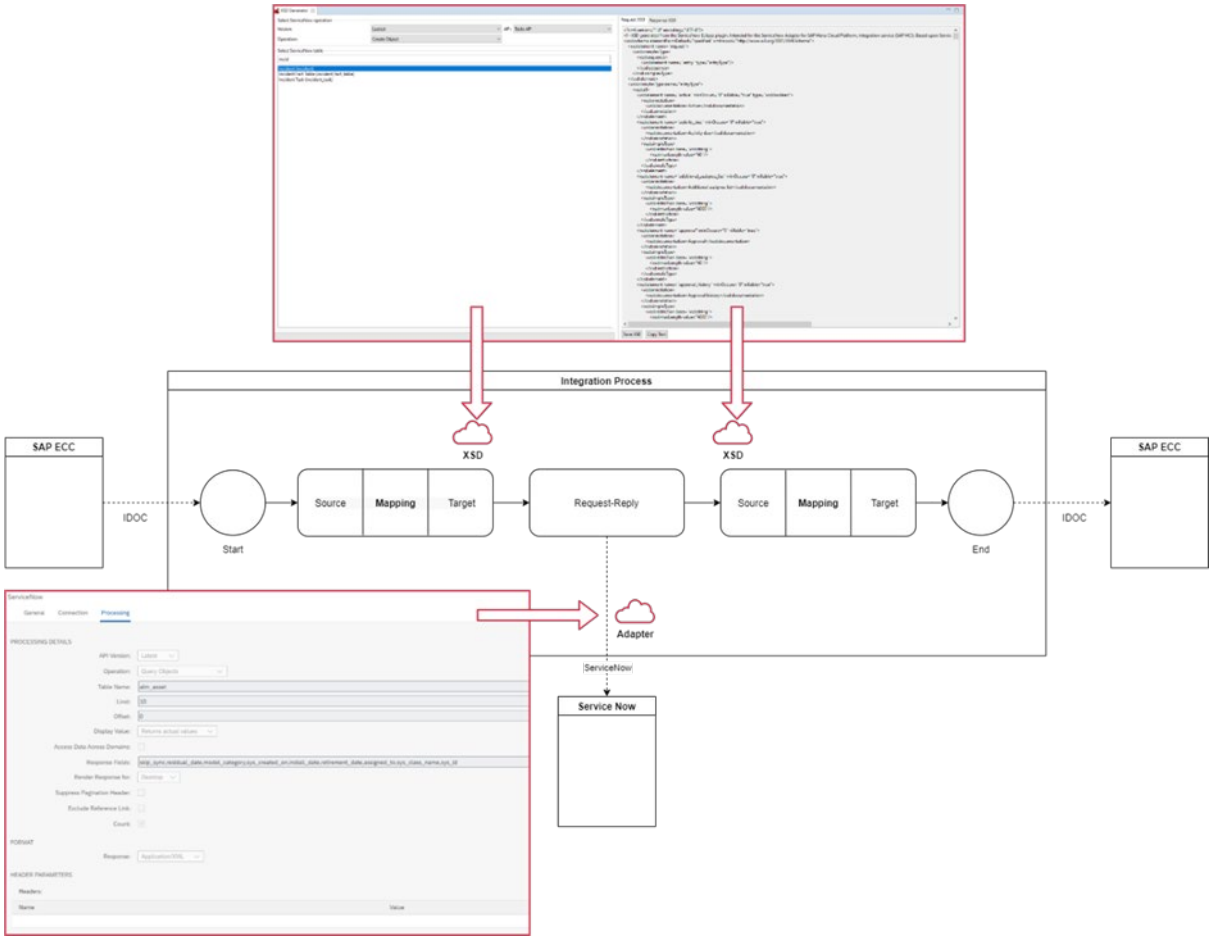

Figure 2.1 ServiceNow Adapter Solutions for SAP Cloud Integration

Figure 2.1 demonstrates how the Request-Reply Step can use the ServiceNow Adapter. This allows the Adapter to connect and interact with ServiceNow and invoke its different operations.

Along with the Adapter, it is possible to use an Eclipse Plug-in. This is delivered together with the Adapter to help the user generate the XSDs of the different tables or business objects that are part of a ServiceNow tenant. The XSDs generated by the Eclipse Plug-in can be imported to an Integration Flow and used in mappings. For instance, in the mapping in front of the Request-Reply Step. The correct XML Request Message to create or update an entity in ServiceNow (see the Mapping Step on the left side of the Request-Reply Step) can be configured. Similarly, the XSDs can also be used to create a mapping that handles the Response Message sent by ServiceNow.

Note: The Eclipse Plug-in is not mandatory. It is simply a facilitator to perform the mapping. Refer to Section 6 for further explanation.

3. SUPPORTED OPERATIONS AND VERSIONS

Currently, the API versions supported are V1, V2, and Latest. Table 1 lists the different operations supported.

Operation	Description
Create Object	This operation creates a table record.
Delete Object	This operation deletes an existing table record.
Get Object by ID	This operation retrieves the details of a ServiceNow record using an ID.
Modify Object	This operation updates existing table record(s), identified by the mandatory sys_id field. **Note:** This operation updates the entire record.
Query Object	This operation retrieves data from ServiceNow based on various filter criteria and query parameters to be specified in the Adapter.
Query Object (Advanced)	This operation retrieves data based on a ServiceNow Query.
Update Object	This operation updates existing table record(s), identified by the mandatory sys_id field. **Note:** This operation partially updates the record based on the input data. It will only update properties mapped in the request body.
Attachment – Delete	This operation deletes an attachment of a specific ServiceNow record.
Attachment – Query Metadata	This operation returns the metadata of multiple attachments.
Attachment – Retrieve Content	This operation retrieves a specific attachment of a ServiceNow record identified by a sys_id value.
Attachment – Retrieve Metadata	This operation retrieves the metadata of an attachment file associated with a ServiceNow record.
Attachment – Upload Binary	This operation uploads a binary file to an existing ServiceNow record.

Table 1 List of Supported ServiceNow Operations

Note: At the time of publication, for a lot of ServiceNow tables, the Update Object and Modify Object operations behave in the same manner. This means that only the provided fields are updated.

For a more detailed explanation of each of the above operations and how to configure them, refer to Section 8.2.

4. ADAPTER INSTALLATION

Refer to the **ServiceNow Adapter and Plug-in Installation Guide** document included in the package for detailed steps on installing the ServiceNow Adapter.

5. ECLIPSE PLUG-IN INSTALLATION

Refer to the **ServiceNow Adapter and Plug-in Installation Guide** document included in the package for detailed steps on installing the Eclipse Workbench/Plug-in and configuring it to connect to ServiceNow.

6. USING THE ECLIPSE PLUG-IN

The Eclipse plug-in or Workbench can be used to extract and generate XSDs of the different ServiceNow tables. These XSDs can then be used in mappings for the Request or Response Messages in integration scenarios. The next sections will explain how to use it.

The ServiceNow XSD Generator is developed to support mapping operations in integration scenarios. The current version of this plug-in supports the generation of Request and Response XSDs for ServiceNow Table and Attachment APIs. The XSDs are generated for specific table objects and operations (example: Create, Modify, Read, Delete, etc.).

After configuring the login information in the preference page, switch to the XSD perspective view in the Eclipse Workbench.

To do so, click on the Window option at the Eclipse menu bar, select Perspectives, and then select Other, as seen in Figure 6.1**Error! Reference source not found.**.

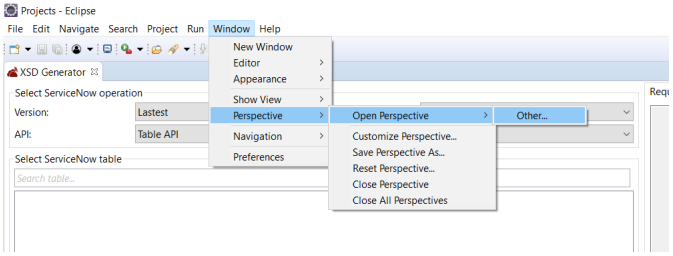

Figure 6.1 Select ServiceNow Perspective

Then select the ServiceNow XSD Generator to get a view similar to Figure 6.2**Error! Reference source not found.**. On the top left side of the view, the Select ServiceNow operation panel is visible. Define the following:

Version: Specify the version of the ServiceNow API to be used. By default, the latest version is selected.
API: Specify the ServiceNow API to connect to. Currently, both the Adapter and the Workbench are working with ServiceNow Tables.
Display Value: Specify the value(s) to be returned in the response. These can either be actual values, display values, or both. The actual value represents the raw value in the database. Display value is a user-understandable value that is shown in the User Interface. When both are selected, ServiceNow returns both display and actual values in the response.
Operation: Specify the ServiceNow operation such as: Create Object, Delete Object, Get Object by ID, Modify Object, Query Objects, Query Objects (Advanced), Update Object, and the following Attachment operations: Delete, Query Metadata, Retrieve Content, Retrieve Metadata, and Upload Binary.

Below the “Select ServiceNow table” text, a panel that allows the selection of a ServiceNow table is available. It is possible to filter and search a ServiceNow table by typing in the search box. See Figure 6.2**Error! Reference source not found.**.

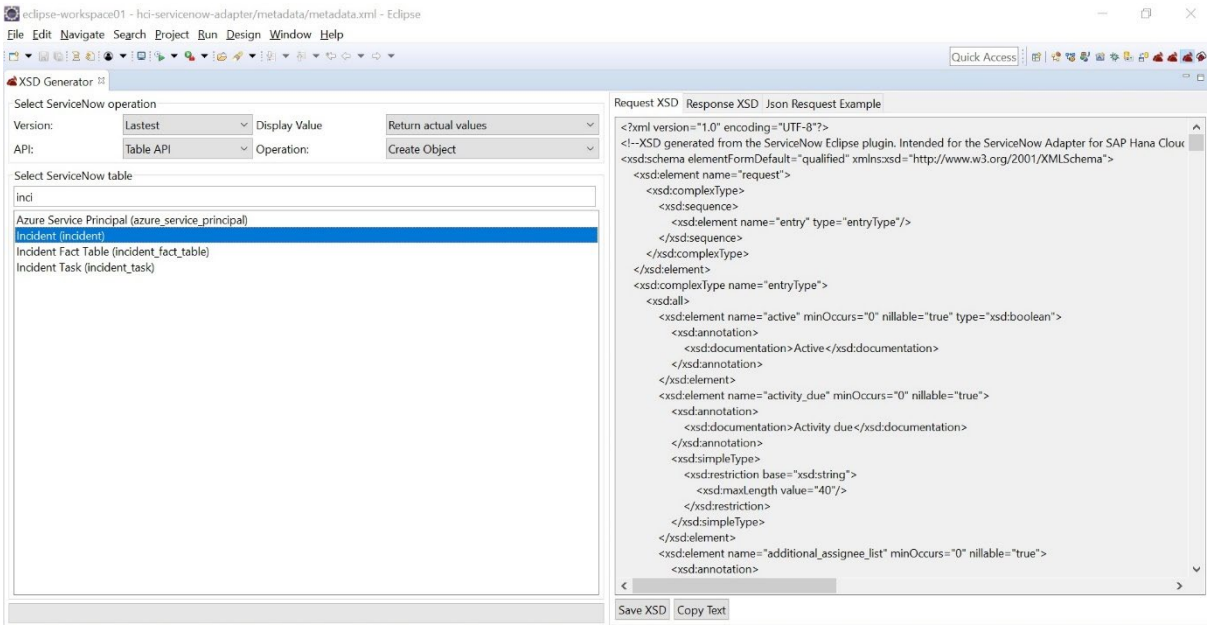

Figure 6.2 ServiceNow XSD Generator

On the top right side of the view, the following tabs are available:

Request XSD: This contains the XSD for the request toward ServiceNow. Note that Request schemas are only available for the Create Object, Update Object, Modify Object, and Attachment – Upload Binary operations. The generated XSD can be saved and imported to an integration flow.
Response XSD: Contains the XSD for the response XSD returned by ServiceNow. Response schemas are available for all operations except Delete.
JSON Request Example: Contains a JSON structure for requests toward ServiceNow. This structure should be used as a reference and modified to fit needs. This example structure is available for the Create, Update, and Modify operations.

To generate an XSD for a specific Table object in ServiceNow, follow the steps below:

1. Select the API version in the **Version** dropdown.
2. Select the API in the **API** dropdown. Currently, the Table API is supported.
3. Select the **Operation** in the dropdown.
4. Once the operation is selected, the ServiceNow table objects available in the tenant will be populated in the bottom left table. The “Search Table…” text box can be used to filter the tables by the names, and the matched results will be shown.
5. Save the generated XSD by clicking the **Save XSD**.

Note: Not all operations require a Request and Response XSD.

Table 1 gives an overview of which operations the Request/Response XSDs are provided by the XSD generator.

Operation	Request	Response
Create Object	x	x
Delete Object		
Get Object by ID		x
Modify Object	x	x
Query Objects		x
Query Objects (Advanced)		x
Update Object	x	x
Attachment – Delete	x	
Attachment – Query Metadata		x
Attachment – Retrieve Content	x	x
Attachment – Retrieve Metadata	x	x
Attachment – Upload Binary	x	x

Table 1 Connection Tab Description

7. AUTHENTICATION

This section details different authentication mechanisms supported by the Adapter. It explains how to create its respective security artifacts in SAP Cloud Integration.

To follow the best practices and the security policies of SAP Cloud Integration, the Adapter works with the standard security artifacts such as Secure Parameter and User Credentials. This way, User, Password, and Secrets keys can be accessed safely by the Adapter via aliases.

7.1 Supported Authentication Mechanisms

The ServiceNow Adapter supports two authentication mechanisms: Basic Authentication and OAuth2 Client Credentials. Basic Authentication requires a user credential in Security Materials to be created. Refer to Section 7.2for more information on how to create a user credential.

Note: The user credential will consist of a username and password.

The OAuth2 Client Credentials require two secure parameters to be created: one for storing Client ID and another for Client Secret.

Refer to Section 7.3for more details on how to create a client secret.

7.2 Creating User Credential in Security Material

In the SAP Cloud Integration tenant, go to Operations View and select Security Material under the Manage Security section.

In Security Material, click on the Create dropdown at the top right and select User Credential. Provide a name and select User Credential in the Type dropdown. Fill in the ServiceNow username and password. Click on Deploy.

7.3 Creating Secure Parameter in Security Material

In Security Material, click on the Create dropdown at the top right and select Secure Parameter. Provide a name for Client ID Alias and Client Secret Alias. Click on Deploy.

For details on how to obtain a Client Secret/Client ID in ServiceNow, refer to Section **Error! Reference source not found.**.

7.4 Configuration of Authentication in the Adapter

OAuth2 Client Credentials (Client ID Alias and Client Secret Alias) are two secure parameters created in Section 7.3. As shown in Figure 0.1, these should be specified alongside the credentials created in Section 7.27.2.

For a more detailed explanation of how to configure the Connection Tab, please refer to Section 8.1.

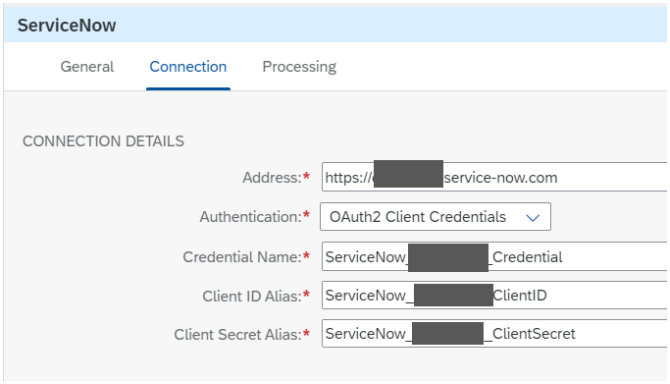

Figure 0.1 Specifying User Credentials and Secure Parameters

8. RECEIVER ADAPTER CONFIGURATION

This section describes the different parameters that can be configured for the Receiver Adapter. This Adapter has the Connection and Processing tabs. These tabs will be explained in the following sections.

8.1 Connection Tab

The Connection tab contains parameters that define how to connect to and how to authenticate against ServiceNow. In case Basic Authentication needs to be used, Basic Authentication will need to be selected for the Authentication field of the **Connection Tab**.

See the example in Figure 0.1.

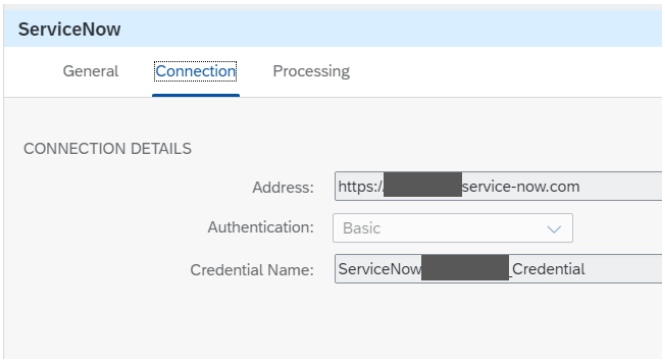

Figure 0.1 Specifying User Credentials in ServiceNow Adapter

For OAuth2 Client Credentials settings, in addition to user credentials, it is needed to specify the OAuth2 Client Credentials. These are the two secure parameters that were created in Section 7.3 (Client ID Alias and Client Secret Alias).

See the example in Figure 0.2.

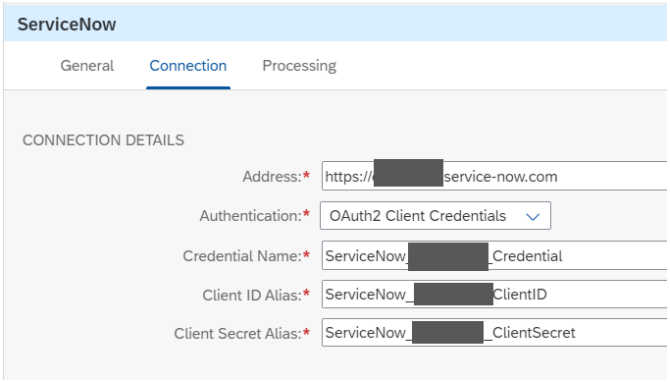

Figure 0.2 Specifying User Credentials and Secure Parameters

Table 3 provides more details of the fields included in the Connection Tab.

Parameter	Description
Address	Specify the recipient's endpoint URL. This value can also be specified dynamically. Example: https://dev12345.service-now.com
Authentication	Select the authentication mechanism to be used while connecting to ServiceNow. It is possible to choose between the following options: • Basic • OAuth2 Client Credentials
Credential Name	Supply the name of the User Credentials artifact that contains the credentials for basic authentication. This represents the ServiceNow credential name (username-password) pair stored as Security Material.
Client ID Alias (only if the Authentication method selected is OAuth2 Client Credentials)	Specify the alias to the security material that contains the Client ID for the endpoint.
Client Secret Alias (only if the Authentication method selected is OAuth2 Client Credentials)	Specify the alias to the security material that contains the Client Secret for the endpoint.

Table 3 Connection Tab Description

Note: The Adapter validates the server certificate while connecting to ServiceNow. In case the root certificate of ServiceNow is not present in the SAP Cloud Integration Keystore, an error will be thrown.

8.2 Processing Tab

In the **Processing Tab**, indicate the type of operation to perform. Likewise, state the details of the request. Figure 0.3 and Figure 0.4 show some sections of the Processing Tab.

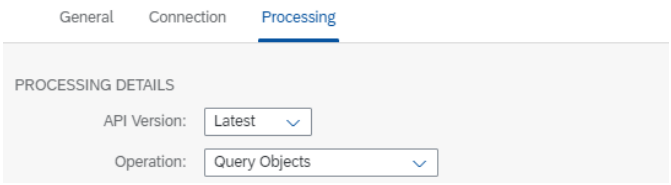

Figure 0.3 API Version

Figure 0.4lists the available operation modes.

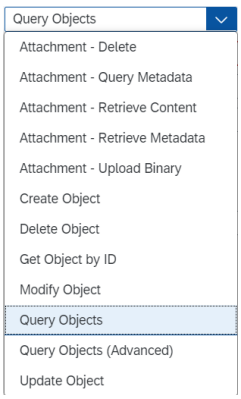

Figure 0.4 List of Operations

Table 4 provides more details of the fields included in the Processing Tab.

Parameter	Description
Processing Details	
API Version	Select the version of the ServiceNow API to be used while calling ServiceNow. Select one of the following versions: Latest (The Adapter automatically selects the latest version of the API). V1 V2
Operation	To access and exchange data with ServiceNow, select the desired operation. It is possible to choose between the following operation options: • Create Object • Delete Object • Get Object by ID • Modify Object • Query Objects • Query Objects (Advanced) • Update Object • Attachment – Delete • Attachment – Query Metadata • Attachment – Retrieve Content • Attachment – Retrieve Metadata • Attachment – Upload Binary
Table Name (only if Attachment-Upload Binary, Create, Delete, Get Object by ID, Modify, Query Objects, Query Objects (Advanced), or Update Object by ID operation is used)	Specify the technical name of the ServiceNow table. Example: incident.
System ID (only if Attachment- Delete/ Retrieve Content/ Retrieve Metadata/ Upload Binary/ Delete, Get Object by ID, Modify, or Update operation is used)	The unique ID of the record. Specifies the unique ID that represents the instance of the record specified in the Table Name field. Also known as sys_id in ServiceNow.
File Name (only if Attachment – Upload Binary operation is used)	Specify the name of the file to be uploaded. The value can also be read dynamically. Example: laptop.png
Encryption Context (only if Attachment – Upload Binary operation is used)	Specify a pre-configured encryption context applied to the file to be uploaded. The value can also be read dynamically. **Note:** Attached file is not encrypted with any encryption context by default.
Display Value (only if Create, Get Object by ID, Modify Object, Query Objects. Query Objects (Advanced), or Upload Object operation is used)	Select if the response returned by ServiceNow should be in display, actual values, or both forms.
Access Data Across Domains	Select if the data should be accessed across domains or not.

(only if Create, Get Object by ID, Modify Object, Query Objects. Query Objects (Advanced), or Upload Object operation is used)	
Response Fields (only if Create, Get Object by ID, Modify Object, Query Objects. Query Objects (Advanced), or Upload Object operation is used)	Specify the fields or properties to be returned in the ServiceNow response. Use commas to separate multiple fields. **Note:** If left empty, all fields/properties are returned.
Limit (only if enabled for the following operations: Attachment – Query Metadata, Query Objects, or Query Objects Advanced)	Limit to be applied on pagination. The default value is 1000.
Offset (only if enabled for the operation Query or Query Advanced)	Offset to be applied on pagination. The default value is 0.
Render Response for	Select if the response should be rendered according to the specified UI view **Note:** This is overridden by Response Fields. Possible values include: Desktop Mobile Both The default value is set to Both.
Suppress System Fields Auto Generation (only if Create, Modify Object, or Upload Object operation is used)	Select to suppress the auto-generation of system fields. The default value is *false*.
Suppress Pagination Header (only if enabled for the operation Query or Query Advanced)	Select to remove the *Link header* from the response. The Link header allows the requesting of additional pages of data when the number of records matching the query exceeds the query limit.
Display Raw Value (only if Create, Modify Object, or Upload Object operation is used)	Select to make the returned ServiceNow record display raw values.
Exclude Reference Link (only if Create, Get Object by ID, Modify Object, Query Objects. Query Objects (Advanced), or Upload Object operation is used)	Select to suppress and remove additional information provided for reference fields. These references include URLs to other resources.
Count (only if enabled for the operation Query or Query Advanced)	Select to execute a select count(*) on the table. **Note**: The default count is unchecked.
Query (only if the Query Advanced operation is used)	Specify a query to represent a complex filter on a list of records manually. Example: unit_name=day^ORDERBYvalue.

Format	
Request (only if the Attachment – Upload Binary, Create, Modify Object, or Update Object operation is used)	Select the format of the request message towards ServiceNow. Possible values include: Application/JSON Application/PDF Application/Text Application/XML Application/Zip Audio/MPEG Image/JPEG Image/PNG Text/CSV Text/Plain Text/XML The default value is Application/XML.
Response	Select the format of the response message to be returned by the searched ServiceNow record. Possible values include: • Application/XML • Application/JSON • Text/XML The default value is Application/XML.
Header Parameters	
Name	Specify the name of the header Parameter.
Value	Specify the value of the header Parameter.
Filters (only if Query Objects operation is selected)	
Query String	
Name	Specify the name of the Query Strings to be used for filtering the data to be returned by the searched ServiceNow record.
Function	Specify the function to be used for the filter. Several functions are provided: Equal Not Equal Starts with Ends with Contains Does not contain Is empty Is not empty Less than Greater than Less than or is Greater than or is Between Is anything Is same Is different Greater than field Less than field Greater than or is field Less than or is field

	Changes Changes from Changes to Is one of Is not one of The default value for this field is Equal.
Value	Specify the value of the Query Strings.
Condition	In case multiple filters need to be used, select the condition to be used for joining the different filters. Possible values include: OR AND The default value is OR.
Query Parameters	
Name	Specify the name of the Query Parameter.
Value	Specify the value of the Query Parameter.
Order By	
Name	Specify if the name of the field is to be used for sorting the results.
Type	Select if the result needs to be sorted using the ascending or descending order. Possible values include: • Ascending • Descending The default value is Ascending.

Table 4 Processing Tab Description

8.2.1 Create Object

This operation creates a new record in a specified ServiceNow Table. The request payload provides the property and field values in the request data. The response body will contain the unique identifier (ID) of the created record if the call is successful.

Figure 0.4 displays an example of how to configure the Adapter to create a new Task in ServiceNow.

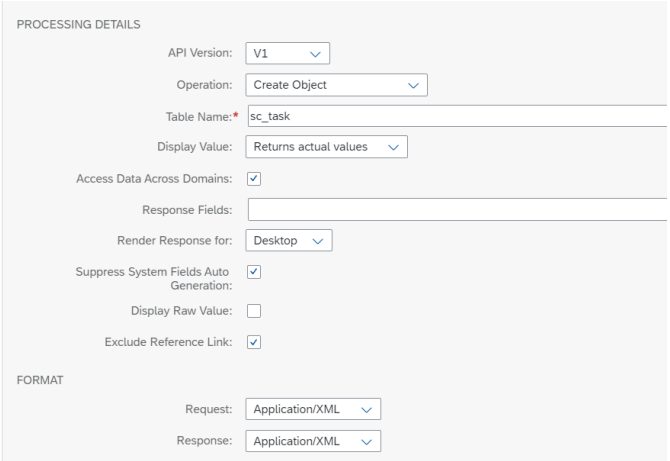

Figure 0.4 Create Object – Processing Details

As shown in Figure 0.4, it is required to specify the Table Name of the object to be created. There are various tables in ServiceNow, such as Incident, User, sc_task, etc. As shown in the Format Section of

Figure 0.4, it is possible to specify the format of the message to be sent to ServiceNow using the Request field. It is possible to select between Application/XML and Application/JSON.

In the case of Application/XML, it is possible to use the XSD generated by the Eclipse Workbench for mapping purposes in an integration flow.

Note: It is also possible to discover the structure and fields required for any ServiceNow table (e.g.: sc_task) using the Eclipse Workbench.

To see the full list of possible tables in ServiceNow, refer to the Eclipse Plug-in in Section 6 (see Figure 6.2).

Note: Eclipse Workbench also supplies a sample JSON Request Message that can be used as a basis in the instance that the format Application/JSON is selected.

Example Request Message:

```
<?xml version="1.0" encoding="UTF-8"?>
<request>
   <entry>
      <activity_due>2021-03-29 10:10:10</activity_due>
      <close_notes>close notes are not available</close_notes>
      <description>give description for task here</description>
   </entry>
</request>
```

For details of any other fields in the Processing Tab, refer to Table in Section 8.2.

8.2.2 Delete Object

This operation deletes a ServiceNow Object record. The operation requires the unique identifier of the ServiceNow record to be specified.

Figure 0.5 displays an example of how to configure the Adapter to delete an incident in ServiceNow.

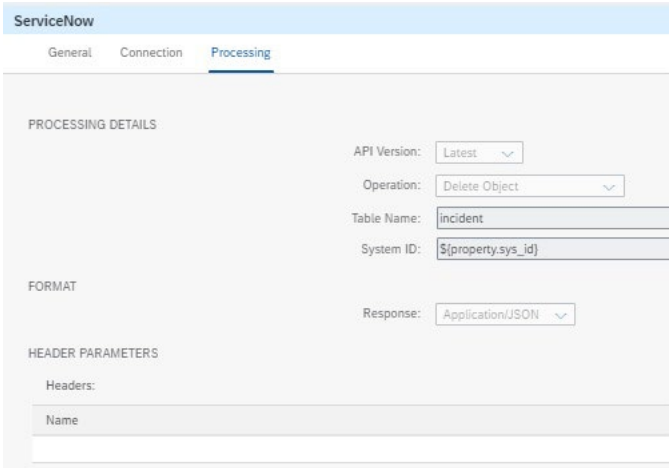

Figure 0.5 Delete Object – Processing Details

As shown in Figure 0.5, the System ID property is needed for the Delete Object operation.

Note: It is possible to specify the ID using a dynamic property or header.

Note: There are no request messages needed for the Delete Object operation.

If the deletion is successful, an OK is returned.

Example Response Message:

{"Result":"OK"}

For details of any other fields in the Processing Tab, refer to Table in Section 8.2.

8.2.3 Get Object by ID

This operation retrieves a record in ServiceNow and its properties. Note that the operation requires the unique ID or system ID of the concerned entity. Figure 0.6 highlights an example of how to configure the Adapter to retrieve details of a specific task in ServiceNow.

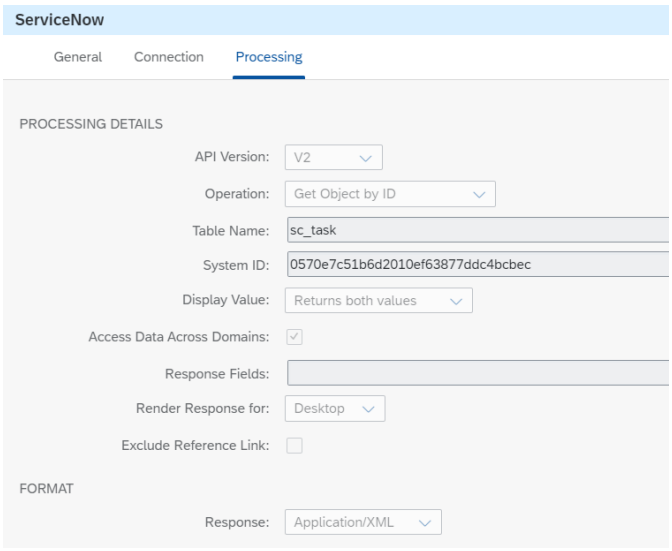

Figure 0.6 Get Object by ID – Processing Details

As shown in Figure 0.6, the System ID property is required for this operation.

Note: It is possible to specify the ID using a dynamic property or header.

Note: There are no request messages needed for the Get Object by ID operation.

Example Response Message:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
   <result>
      <approval>not requested</approval>
      <number>TASK0000001</number>
      <short_description>Order from vendor or move from in-stock inventory
      </short_description>
      <sys_class_name>sc_task</sys_class_name>
      <sys_created_by>admin</sys_created_by>
      <sys_domain>
         <link>https://xxx.service-now.com/api/now/v2/table/sys_user_group/global</link>
         <value>global</value>
      </sys_domain>
      <sys_updated_on>2019-11-29 20:58:46</sys_updated_on>
      <upon_reject>cancel</upon_reject>
      <request>
         <link>https://dev89342.service-
now.com/api/now/v2/table/sc_request/6eed229047801200e0ef563dbb9a71c2</link>
         <value>6eed229047801200e0ef563dbb9a71c2</value>
      </request>
   </result>
</response>
```

For details of any other fields in the Processing Tab, refer to Table in Section 8.2.

8.2.4 Modify Object

This operation modifies a ServiceNow Object record. The request payload should contain the property values that are needed to update a record in ServiceNow based on a specific record ID. The record ID is represented by the System ID property that is mandatory.

Note: It is possible to specify the ID using a dynamic property or header.

The Modify operation is the technical equivalent of the PUT method. This means the specified record is overwritten with the provided data. Normally, PUT overwrites any field that is not provided with a null value.

Note: At the time this document was published for a lot of ServiceNow tables the Modify operation behaves more like a PATCH method. This means that only the properties provided as part of the input payload are updated. The rest of the fields are left intact. This is the same behavior as the Update operation that is explained in Section **Error! Reference source not found.**.

Figure 0.7 showcases an example of how to configure the Adapter to modify the details of a specific image in ServiceNow.

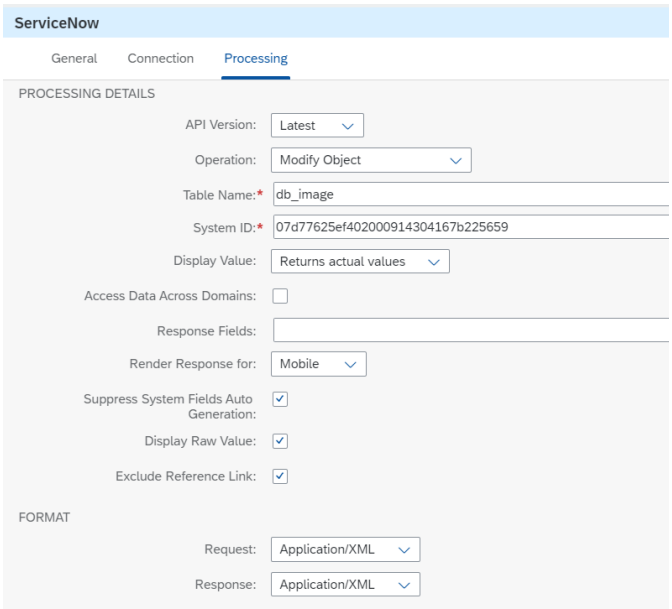

Figure 0.7 Modify Object – Processing Details

Example Request Message:

```
<request>
  <entry>
    <name>design_thumb4.png</name>
  </entry>
</request>
```

Example Response Message:

```
<?xml version="1.0" encoding="UTF-8"?>
<response>
   <result>
      <image/>
      <thumbnail/>
      <size_bytes>39050</size_bytes>
      <format>png</format>
      <sys_mod_count>4</sys_mod_count>
```

```
      <active>true</active>
      <sys_updated_on>2020-12-02 10:24:05</sys_updated_on>
      <sys_tags/>
      <sys_class_name>db_image</sys_class_name>
      <sys_id>07d77625ef402000914304167b225659</sys_id>
      <sys_package>cecc5085e8e100107850694c2d9d2d51</sys_package>
      <sys_update_name>db_image_07d77625ef402000914304167b225659</sys_update_name>
      <sys_updated_by>Admin</sys_updated_by>
      <sys_created_on>2012-02-07 21:33:30</sys_created_on>
      <name>design_thumb4.png</name>
      <width>271</width>
      <sys_name>design_thumb4.png</sys_name>
      <sys_scope>global</sys_scope>
      <category/>
      <sys_created_by>mark.odonnell</sys_created_by>
      <height>201</height>
      <sys_policy/>
   </result>
</response>
```


For details of any other fields in the Processing Tab, refer to Table in Section 8.2.

8.2.5 Update Object

This operation updates an existing ServiceNow Object record. The request payload should contain the property values that are needed to update a record in ServiceNow based on a specific record ID. The update operation is the technical equivalent of a PATCH method. This means the specified record is overwritten with the provided data. Any property that is not present in the input payload is left untouched.

Figure 0.8 highlights an example of how to configure the Adapter to update the details of a task in ServiceNow. The concerned task is identified by a mandatory System ID property.

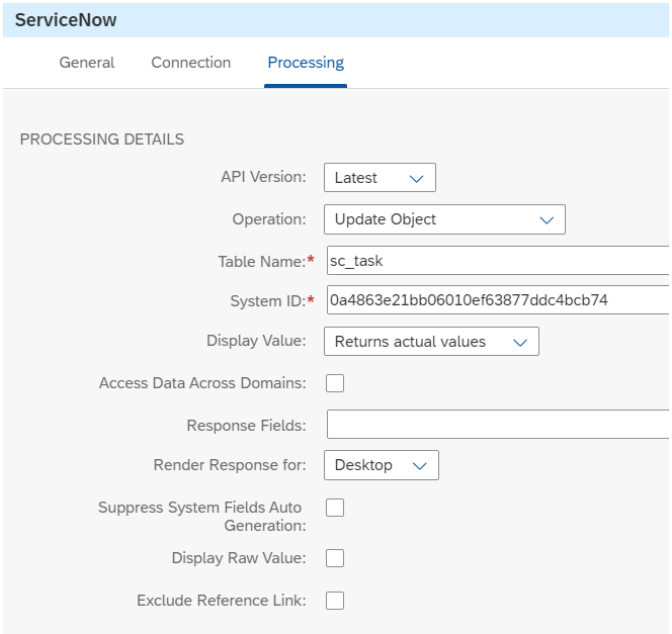

Figure 0.8 Update Object - Processing Details

Note: It is possible to specify the System ID using a dynamic property or header of choice.

Example: ${property.systemID} or ${header.systemID}.

Figure 0.9 showcases another example that uses the ServiceNow Adapter to update an existing incident. In this example, the record to be updated is identified using the **System ID:** 46e18c0fa9fe19810066a0083f76bd56.

Note: System ID represents the ServiceNow's sys_id.

The example expects the Request Message to be XML and requests ServiceNow to return the response as XML. These are specified by the Request and Response properties under the Format section in Figure 0.9.

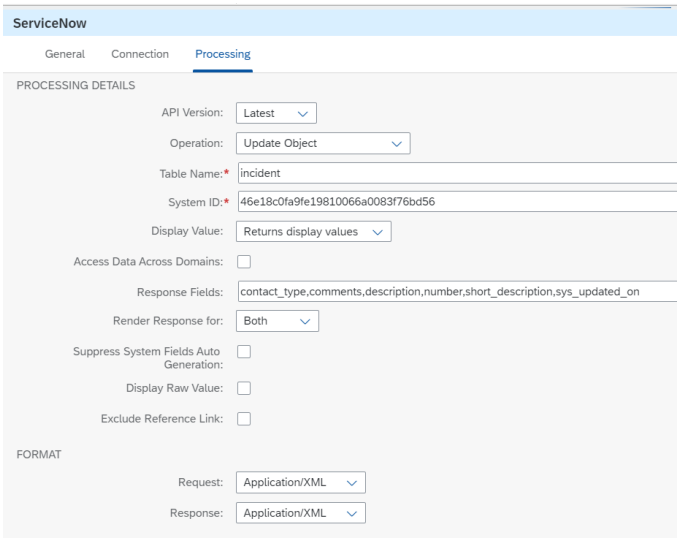

Figure 0.9 Update Object – Processing Details

Example Request Message:

```
<request>
        <entry>
                <short_description>Home2</short_description>
                <comments>test</comments>
        </entry>
</request>
```

Example Response Message:

```
        <?xml version="1.0" encoding="UTF-8"?>
                <response>
                        <result>
                                 <comments>2021-03 (Additional comments)  </comments>
                                <contact_type>Phone</contact_type>
                                 <description>  Logged out and back in again and the  problem  was  solved
                                 resolved.</description>
                                <number>INC0000014</number>
                                <short_description>Home2</short_description>
                                <sys_updated_on>2021-03-10 05:43:11</sys_updated_on>
          </result>
        </response>
```

For details of any other fields in the Processing Tab, refer to Table in Section 8.2.

8.2.6 Query Objects

This operation retrieves records in ServiceNow and their properties. There is no request payload needed for the operation, but the Table name is required. To query data from ServiceNow, it is possible to specify the query using properties such as Filter (see Section For details of any other fields in the Processing Tab, refer to Table in Section 8.2.
8.2.6.1), Query Parameters (see Section 8.2.6.2), and Order by (see Section 8.2.6.3).

Figure 0.10 displays an example of how to configure the Adapter to Query the details of the DB_Image table in ServiceNow.

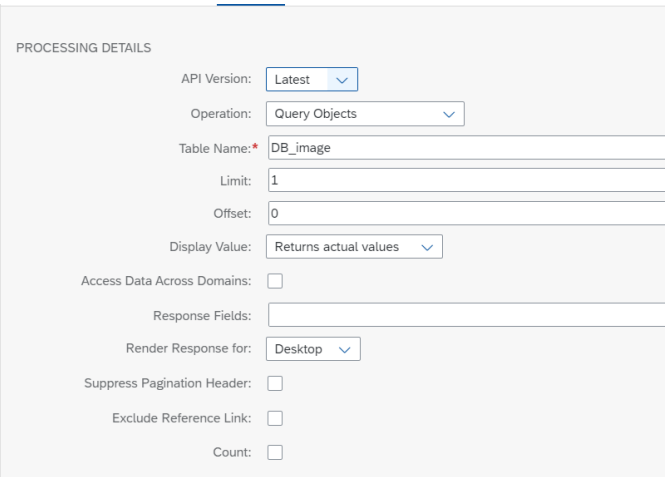

Figure 0.10 Query Objects - Processing Details

Note attributes such as Limit, Offset and Count.

For details of any other fields in the Processing Tab, refer to Table in Section 8.2.

8.2.6.1 Query String

It is possible to add as many filters as needed by clicking the **Add** button using the Query String table specified under the Filter section.

See below the properties that need to be added for each row:

Name: Type the name of the field to filter. Example: Active.
Function: Select the operation function. Example: Equal.
Value: Type the value to be used in the function. Example: True.
Condition: In case of more than one filter, select the corresponding condition: OR / AND.

8.2.6.2 Query Parameters

Query Parameters are complex filters on a list of records. It is possible to add as many query parameters as needed by clicking the **Add** button. It is possible to create a query by filling in the name of the parameter (or field) and the value to filter by.

Example: active = true OR category = network.

See below the properties that need to be added for each row:

Name: The name of the field in ServiceNow. Example: **Active** in the incident table.
Value: The value to filter records by. Example: **True**.

8.2.6.3 Order By

The *Order by Parameters* action sorts the result set returned by ServiceNow based on the provided specification. As many parameters can be added as needed by clicking the **Add** button.

See below the properties that need to be added for each row:

Name: Type the name of the field to be ordered.
Type: Select the type of sort by Ascending or Descending.

For details of any other fields in the Processing Tab, refer to Table in Section 8.2.

8.2.6.4 Sample Query Objects Scenario

The example in this section will explain how to configure the ServiceNow Adapter to query for assets using filters. To achieve that, continue as follows:

After filling in the authentication method and credentials in the **Connection Tab**, go to the **Processing Tab** to configure the request.
Select **Latest** as the **API Version**, **Query Objects** as the **Operation,** and type **alm_asset** in the **Table Name**. Assume that we already know there are 52 records in the Asset table and that only the latest 10 records are needed. Since a maximum of 10 records are desired to be returned, it is required to specify an **Offset** of 42 (starting point), and a **limit** of **10** records (43-52).
Specify the **Response fields** that will be returned by ServiceNow as part of the response. This is a comma-separated list of fields.

Example: **No suggestions, cost,sys_id, install_status**

Select the **Count** checkbox to retrieve the total number of assets.
Specify the **Response** format to be **Application/JSON**. No Header Parameters will be applied.
Specify Query Strings to fit filter criteria. In the example, two filters were specified: po_number ends with 00003 OR sys_id starts with 00aba08d3.
Specify the Query parameters to meet needs. In this case, only records with an **install_status** of **1** are needed.
Lastly, order the returned records in the response by **serial_number** in **descending** order.

Figure 0.11 and Figure 0.12 showcase an example of how to configure the Adapter with the details specified in the above steps.

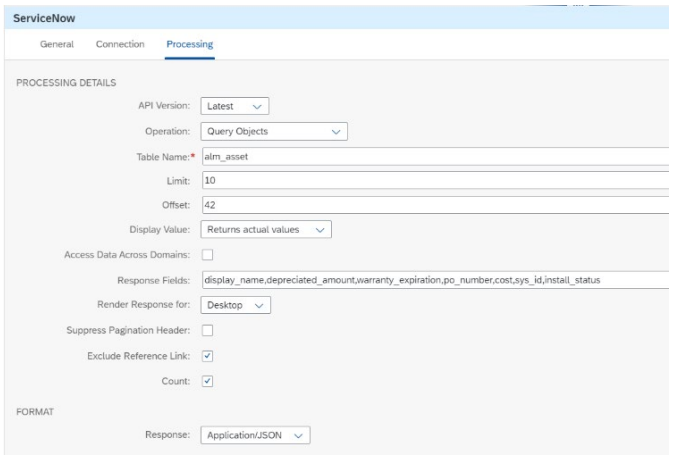

Figure 0.11 Query Assets with Filters- Part 1

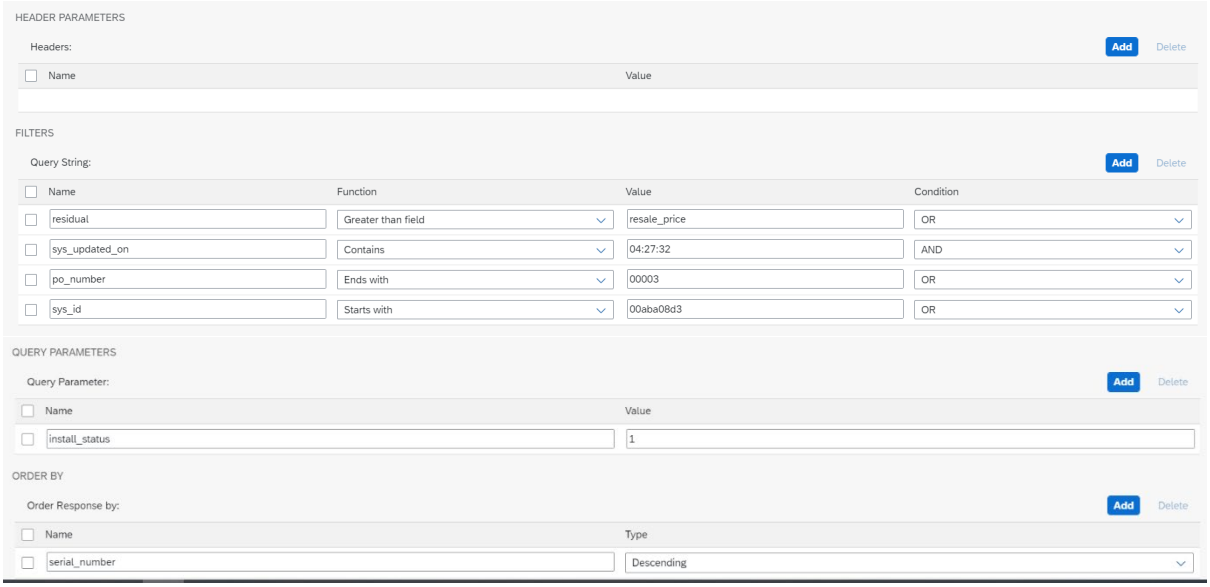

Figure 0.12 Query Assets with Filters - Part 2

Note: No Request Message/Payload is required for the Query Objects operation. After running the integration flow, a result similar to the one presented below is returned.

Example Response Message:


```
{
   "result": [
      {
         "sys_id": "daa9a80d3790200044e0bfc8bcbe5dcd",
         "po_number": "PO100003",
         "depreciated_amount": "1985.21",
         "cost": "2499.99",
         "install_status": "1",
         "display_name": "P1000761 - Apple MacBook Pro 17\"",
         "warranty_expiration": "2020-03-19"
      },
      {
         "sys_id": "1ea9a80d3790200044e0bfc8bcbe5dc9",
         "po_number": "PO100003",
         "depreciated_amount": "1986.57",
         "cost": "2499.99",
         "install_status": "1",
         "display_name": "P1000885 - Apple MacBook Pro 17\"",
         "warranty_expiration": "2020-03-18"
      },
      {
         "sys_id": "c7a9a80d3790200044e0bfc8bcbe5df6",
         "po_number": "PO100003",
         "depreciated_amount": "1935.92",
         "cost": "2499.99",
         "install_status": "1",
         "display_name": "P1000853 - Apple MacBook Pro 17\"",
         "warranty_expiration": "2020-04-24"
      },
      {
```

```
    "sys_id": "63a9280d3790200044e0bfc8bcbe5d45",
    "po_number": "PO100003",
    "depreciated_amount": "1236.36",
    "cost": "1599.99",
    "install_status": "1",
    "display_name": "P1000617 - Apple MacBook Air 13\"",
    "warranty_expiration": "2020-04-27"
},
{
    "sys_id": "08a96c0d3790200044e0bfc8bcbe5da3",
    "po_number": "PO100003",
    "depreciated_amount": "1413.57",
    "cost": "1799.99",
    "install_status": "1",
    "display_name": "P1000556 - Apple MacBook Pro 15\"",
    "warranty_expiration": "2020-04-04"
},
{
    "sys_id": "bda9204d3790200044e0bfc8bcbe5dee",
    "po_number": "PO100003",
    "depreciated_amount": "1038.6",
    "cost": "1329",
    "install_status": "1",
    "display_name": "P1000302 - Dell Inc. Precision T5500 Workstation",
    "warranty_expiration": "2020-04-11"
},
{
    "sys_id": "5ea9a80d3790200044e0bfc8bcbe5dc5",
    "po_number": "PO100003",
    "depreciated_amount": "1998.9",
    "cost": "2499.99",
    "install_status": "1",
    "display_name": "P1000817 - Apple MacBook Pro 17\"",
    "warranty_expiration": "2020-03-09"
},
{
    "sys_id": "61a9204d3790200044e0bfc8bcbe5d65",
    "po_number": "PO100003",
    "depreciated_amount": "1064.84",
    "cost": "1379",
    "install_status": "1",
    "display_name": "P1000288 - Dell Inc. Precision T3500 Workstation",
    "warranty_expiration": "2020-04-28"
},
{
    "sys_id": "bfa96c0d3790200044e0bfc8bcbe5d8b",
    "po_number": "PO100003",
    "depreciated_amount": "1434.27",
    "cost": "1799.99",
    "install_status": "1",
    "display_name": "P1000495 - Apple MacBook Pro 15\"",
    "warranty_expiration": "2020-03-14"
```

```
    },
    {
      "sys_id": "04a96c0d3790200044e0bfc8bcbe5db3",
      "po_number": "PO100003",
      "depreciated_amount": "1404.7",
      "cost": "1799.99",
      "install_status": "1",
      "display_name": "P1000503 - Apple MacBook Pro 15\"",
      "warranty_expiration": "2020-04-13"
    }
  ]
}
```


Note: As the Count checkbox was ticked, a Header Parameter is returned. It is named **Header: X-Total-Count** in the response.

Figure 0.13 shows an instance when the Count checkbox was ticked and a total of 52 records matched the query.

Figure 0.13 Header: X-Total-Count

8.2.7 Query Objects (Advanced)

Like the Query operation, the Query Objects (Advanced) operation filters data from ServiceNow by specifying an encoded query.

Note: Compared to the Query Objects operation, Query Objects (Advanced) enables developers familiar with ServiceNow queries to use complex queries.

The query follows the syntax: *<col_name><operator><value>.*

Where:

<col_name> is the name of the table column to filter against,
<operator> is the supported operators are listed in Table 5, and
<value> is the value to match against.

Operator	Description	Example
=	The provided <col_name> needs to be an exact match of the <value>.	active=true
!=	The provided <col_name> should not match <value>.	category!=database
^	Logically AND multiple query statements.	active=true^category=database
^OR	Logically OR multiple query statements	active=false^ORcategory=database
LIKE	The provided <col_name> contains the specified string. Note that this operator only works for <col_name> fields with the data type being String.	categoryLIKEdata

STARTSWITH	The provided <col_name> starts with the specified string. Note that this operator only works for <col_name> fields with the data type being String.	depreciation_dateSTARTSWITH2017-0
ENDSWITH	The provided <col_name> ends with the specified string. Note that this operator only works for <col_name> fields with the data type being String.	depreciation_dateENDSWITH2017-0

Table 5 Supported Query Operators

Note: All the parameters mentioned in Table 5Table are case-sensitive.

Note: A query can be made of multiple entries.

Example: depreciation_dateSTARTSWITH2017-0^ORDERBYserial_number^COUNT=true

Figure 0.14 highlights an example of how to configure the Adapter to query the details of the asset (alm_asset) table in ServiceNow.

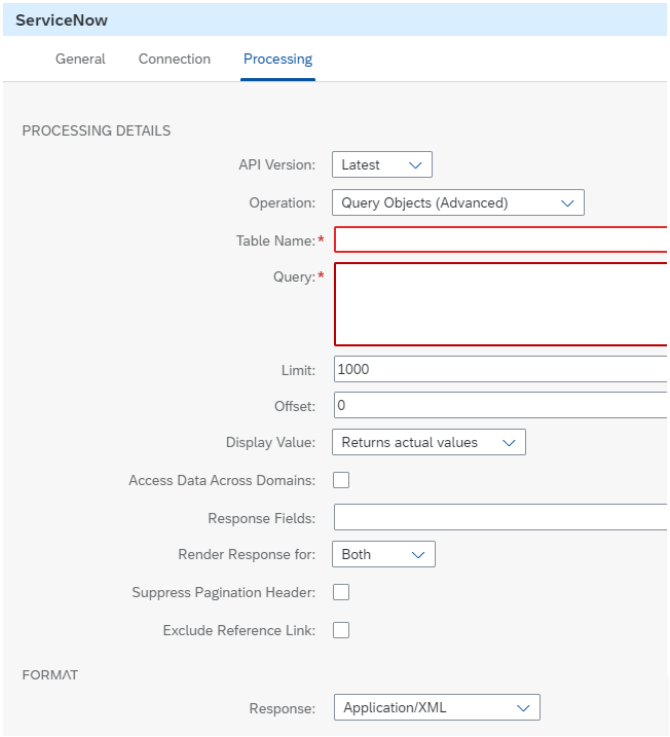

Figure 0.14 Query Objects (Advanced) – Processing Details

For details of any other fields in the Processing Tab, refer to the summary Table in Section 8.2.

8.2.7.1 Sample Query Objects (Advanced) Scenario

The example in this section explains how to configure the ServiceNow Adapter to query objects using the Query Objects (Advanced) operation. To achieve that, proceed as follows:

After supplying the authentication method and credentials in the **Connection** Tab, proceed to the **Processing** tab to configure the request.
Select **Latest** as the **API Version**, and **Query Objects (Advanced)** as the **operation.** Also, specify **alm_asset** in the **Table Name**.
In the **Query** textbox, add a query statement to filter records that have a depreciation date starting from 2017-0. The records returned in the response should be sorted in ascending order using the serial_number. Additionally, the total number of records found should be returned in the header.

Example: **depreciation_dateSTARTSWITH2017-0^ORDERBYserial_number^COUNT=true**

Set the **Limit** to **5** to specify the number of records to be returned. Also, set the **Offset** as default **0**. Furthermore, fill in the **Response fields** with a comma-separated list of fields.

Example: **No suggestions, display_name, po_number, serial_number, sys_id**

Lastly set the **Response** format to **Text/XML**.

Leave all the other fields to their default values. Note that this operation does not need a request message. Figure 0.15 displays an example of how to configure the Adapter to Modify the details of a specific image in ServiceNow.

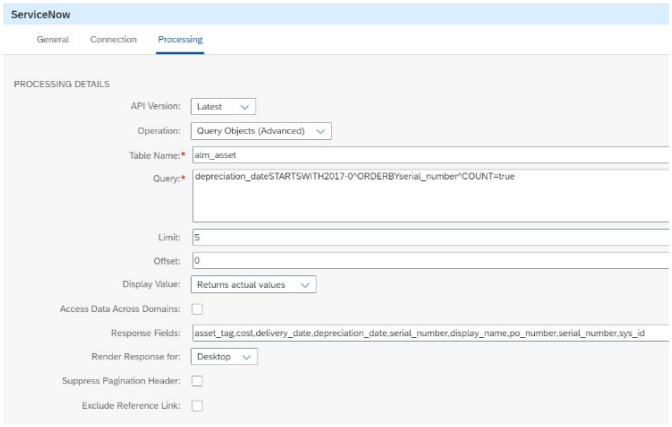

Figure 0.15 Query Objects (Advanced) – Processing Details

After deploying and executing the integration flow, a response similar to the one below is returned.

Example Response Message:

```
<?xml version="1.0" encoding="UTF-8"?>

<response>
   <result>
      <asset_tag>P1000503</asset_tag>
      <cost>1799.99</cost>
      <delivery_date>2017-02-24 08:00:00</delivery_date>
      <depreciation_date>2017-04-15 07:00:00</depreciation_date>
      <display_name>P1000503 - Apple MacBook Pro 15&amp;quot;</display_name>
      <po_number>PO100003</po_number>
      <serial_number>ABE-486-V17263-DO</serial_number>
      <sys_id>04a96c0d3790200044e0bfc8bcbe5db3</sys_id>
   </result>
   <result>
      <asset_tag>P1000620</asset_tag>
      <cost>1599.99</cost>
      <delivery_date>2016-12-15 08:00:00</delivery_date>
      <depreciation_date>2017-01-22 08:00:00</depreciation_date>
      <display_name>P1000620 - Apple MacBook Air 13&amp;quot;</display_name>
      <po_number>PO100001</po_number>
      <serial_number>ADD-886-Q84369-TJ</serial_number>
      <sys_id>c6a9280d3790200044e0bfc8bcbe5d05</sys_id>
   </result>
   <result>
      <asset_tag>P1000515</asset_tag>
      <cost>1799.99</cost>
      <delivery_date>2017-01-23 08:00:00</delivery_date>
      <depreciation_date>2017-03-03 08:00:00</depreciation_date>
      <display_name>P1000515 - Apple MacBook Pro 15&amp;quot;</display_name>
```

```
    <po_number>PO100002</po_number>
    <serial_number>AEG-330-E12514-IC</serial_number>
    <sys_id>73a96c0d3790200044e0bfc8bcbe5d80</sys_id>
  </result>
  <result>
    <asset_tag>P1000540</asset_tag>
    <cost>1799.99</cost>
    <delivery_date>2017-05-16 07:00:00</delivery_date>
    <depreciation_date>2017-07-18 07:00:00</depreciation_date>
    <display_name>P1000540 - Apple MacBook Pro 15&amp;quot;</display_name>
    <po_number>PO100005</po_number>
    <serial_number>AJJ-291-O84969-RM</serial_number>
    <sys_id>18a96c0d3790200044e0bfc8bcbe5de2</sys_id>
  </result>
  <result>
    <asset_tag>P1000406</asset_tag>
    <cost>2499.99</cost>
    <delivery_date>2017-05-16 07:00:00</delivery_date>
    <depreciation_date>2017-09-06 07:00:00</depreciation_date>
    <display_name>P1000406 - Apple MacBook Pro 17&amp;quot;</display_name>
    <po_number>PO100005</po_number>
    <serial_number>APT-932-E53341-UB</serial_number>
    <sys_id>71a9e80d3790200044e0bfc8bcbe5d8c</sys_id>
  </result>
</response>
```


There are a total of 233 records, as shown by the **X-Total-Count header** in the response as seen in Figure 0.16.

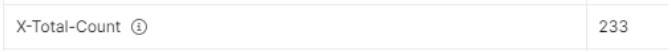

Figure 0.16 Header: X-Total-Count

8.2.8 Attachment – Delete

This operation allows the deletion of an attachment to an existing ServiceNow record. To initiate this operation, the System ID with the attachment to be deleted is required.

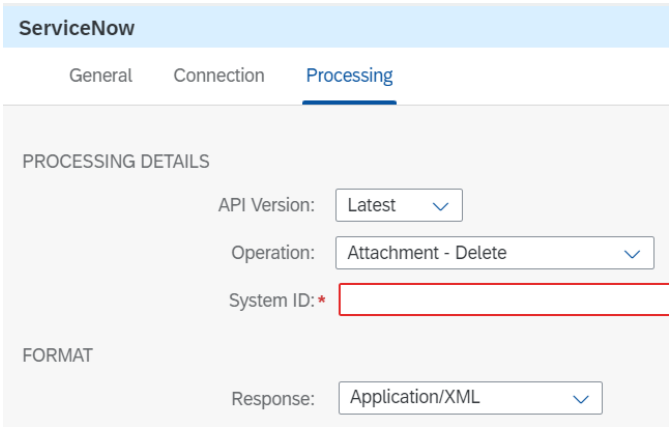

Figure 8.18 Attachment – Delete Processing Details

As shown in Figure 8.18, the System ID property is required for the Delete operation. Note that it is possible to specify the ID using a dynamic property or header.

If the deletion is successful, an OK is returned.

For details of any other fields in the Processing Tab, refer to Table in Section 8.2.

8.2.9 Attachment – Query Metadata

This operation queries the metadata of files attached to existing ServiceNow records.

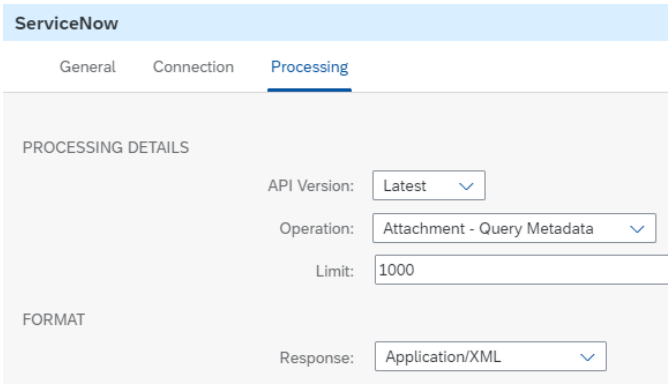

Figure 8.19 Attachment – Query Metadata Processing Details

As seen in Figure 8.19, the default limit to the pagination of results is 1000.

For details of any other fields in the Processing Tab, refer to Table in Section 8.2.

8.2.10 Attachment – Retrieve Content

This operation retrieves the content of a file attached to an existing ServiceNow record.

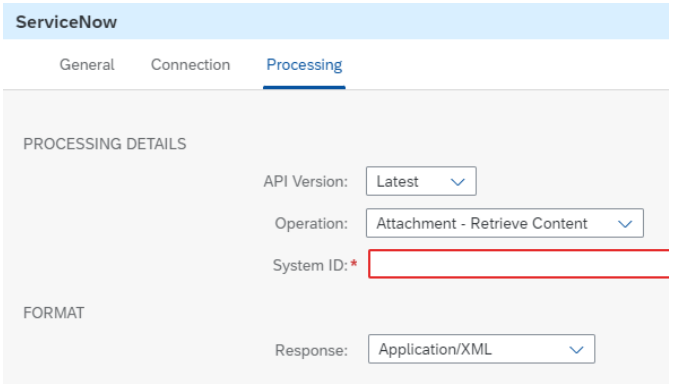

Figure 8.20 Attachment – Retrieve Content Processing Details

As shown in Figure 8.20, the Attachment – Retrieve Content operation requires the System ID of the ServiceNow record with the attachment being retrieved.

Note: This operation does not retrieve the actual attached file to a ServiceNow record.

For details of any other fields in the Processing Tab, refer to Table in Section 8.2.

8.2.11 Attachment – Retrieve Metadata

This operation allows the retrieval of the metadata of a specific ServiceNow record.

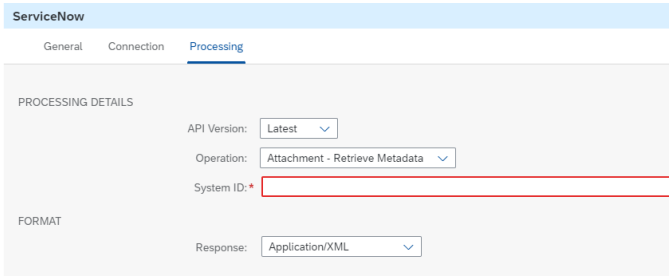

Figure 8.21 Attachment – Retrieve Metadata Processing Details

As seen in Figure 8.21, the System ID of the specific ServiceNow record is needed to retrieve its metadata. The metadata of the ServiceNow record is returned.

Refer to Table 6 for the elements of the returned metadata and their corresponding descriptions.

Element	Description
result.average_image_color	If the file attached to the searched ServiceNow record is an image, the sum of all colors is displayed in RGB or the number of pixels.
result.compressed	This indicates whether the attached file has been compressed or not. Values: true: The file has been compressed. false: The file has not been compressed.
result.content_type	This indicates the content type of the attachment file associated with the ServiceNow record.
result.created_by_name	The full name of the entity that originally created the attachment file is displayed here.
result.download_link	The full name of the entity that originally created the attachment file is displayed here.
result.file_name	The nominated name of the attached file is displayed here.
result.image_height	If the attached file is an image, its height is displayed here. The unit of value is in pixels.
result.image_width	If the attached file is an image, its width is displayed here. The unit of value is in pixels.
result.size_bytes	The size of the attachment is shown here. The unit of value is in bytes.
result.size_compressed	The size of the compressed attachment file is shown here. If the file is not compressed, there is no value displayed.
result.sys_created_by	The name of the entity that originally created the attachment file is shown here.
result.sys_created_on	The date and time when the attachment file was initially uploaded to the ServiceNow record are displayed here.
result.sys_id	This is the sys_id of the attachment file.
result.sys_mod_count	This field shows the total number of times the attachment file has been modified.
result.sys_tags	Any system tags associated with the attachment file are seen here.
result.sys_updated_by	This field displays the name of the entity that last updated the attachment file.
result.sys_updated_on	This shows the date and time that the attachment file was last updated.
result.table_name	The name of the table to which the attachment is associated is shown here.
result.table_sys_id	The sys_id of the table associated with the attachment is displayed here.
result.updated_by_name	The full name of the entity that last updated the attachment file is seen here.

Table 6 Summary of Attachment Metadata Elements

For details of any other fields in the Processing Tab, refer to Table in Section 8.2.

8.2.12 Attachment – Upload Binary

This operation allows the uploading of a binary file to an existing ServiceNow record.

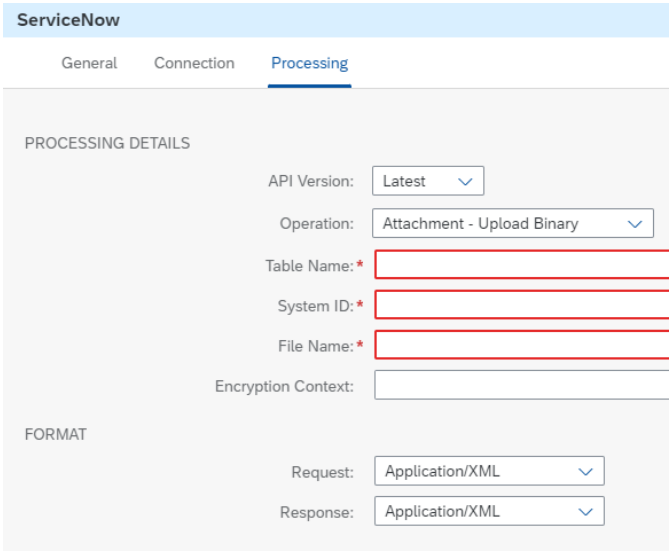

Figure 8.22 Attachment – Upload Binary Processing Details

As indicated in Figure 8.22, attaching a binary file to an existing ServiceNow record requires identifying its Table Name and System ID. Likewise, the File Name of the attachment must also be provided.

For details of any other fields in the Processing Tab, refer to Table in Section 8.2.

9. SAMPLE SCENARIOS EXPLAINED

In this section, a sample integration scenario is discussed to highlight the use of the Adapter and the Eclipse Plug-in.

Figure 9.1 shows an integration flow in SAP Cloud Integration that replicates an account from Microsoft Dynamics to ServiceNow as an AWS account (cmdb_ci_aws_account).

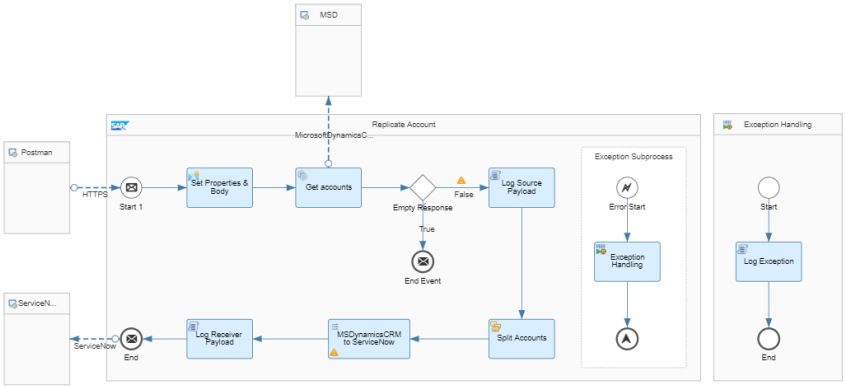

Figure 9.1 Sample Integration Flow

For simplicity, the integration flow exposes an HTTPS endpoint. This means that the integration flow needs to be triggered externally via an HTTPS call.

Using the Microsoft Dynamics CRM (Customer Relationship Management) adapter, the integration flow queries MS Dynamics for all accounts created and modified since the last time the integration flow ran. In case there are no new accounts, the process ends. For this purpose, use the MS Dynamics CRM adapter to query accounts. Use the Query operation and specify Accounts as an entity.

Otherwise, if records are found, the source payload is logged. The returned accounts list is split and mapped from the MS Dynamics account structure to the ServiceNow AWS account structure.

To transform the structure coming from Microsoft Dynamics CRM to the ServiceNow structure, XSDs for the Accounts (Microsoft Dynamics CRM) and cmdb_ci_aws_account (ServiceNow) are needed. The XSD of the data returned by the Microsoft Dynamics CRM needs to be created in the Eclipse Plug-in. How to use the MS Dynamics CRM Adapter is not discussed as its mention is simply for illustration purposes. Any other system can be used.

To generate the XSD needed on the ServiceNow side to create an account, use the Eclipse Plug-in.

See Figure 9.2 for an example of how to set up Workbench.

As shown in Figure 9.2, select **Latest** in the Version, **Table API** in API, **Return actual values** as the Display Value and **Create Object** in the Operation. Note that these settings are similar to the ones configured in the ServiceNow Adapter in Cloud Integration.

By typing the name of the object, the list of tables is filtered. Select the **AWS_Account** table.
The Request XSD will automatically be populated on the right page as seen in Figure 9.2. The generated XSD can now be later imported to SAP Cloud Integration.

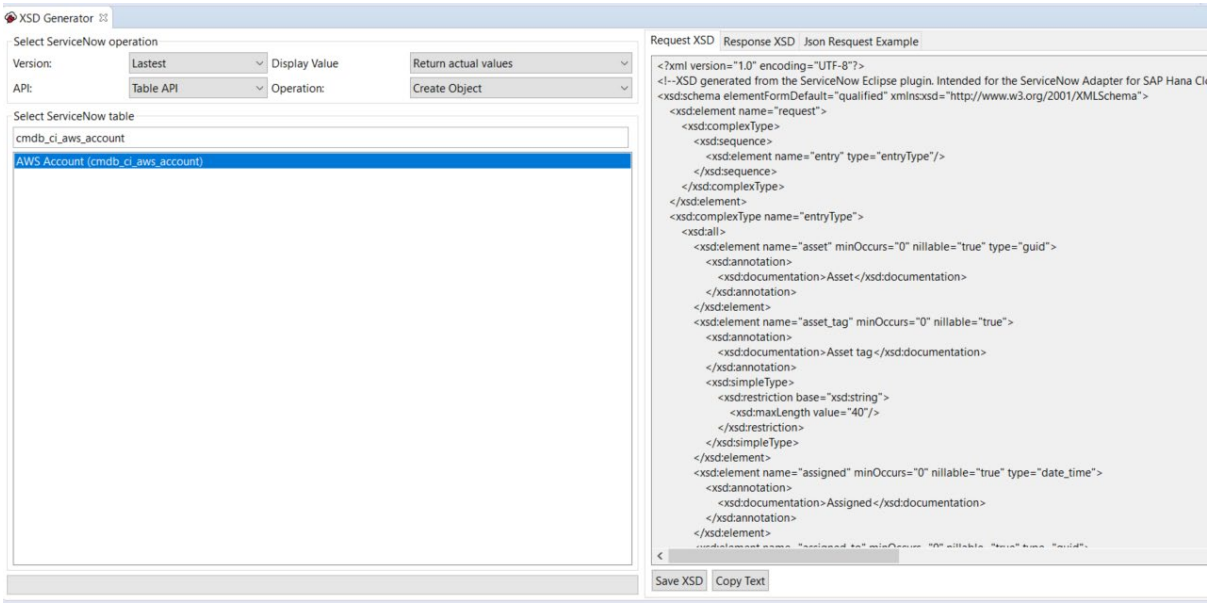

Figure 9.2 Using the Eclipse Workbench to generate the Create XSD

Both schemas can be used for mapping in Cloud Integration. This mapping step transforms the structure from **MS Dynamics CRM** to the one from **ServiceNow**. See Figure 9.3.

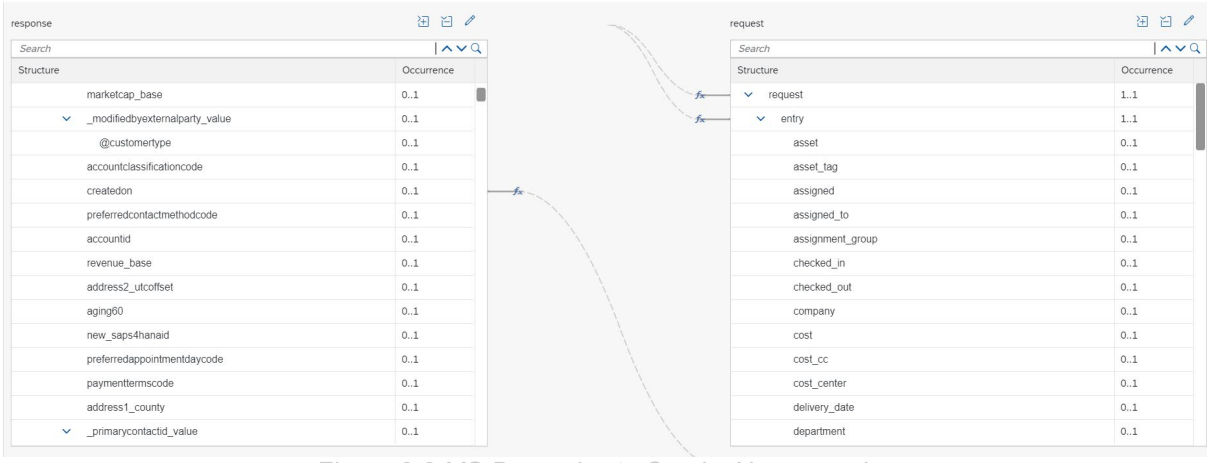

Figure 9.3 MS Dynamics to ServiceNow mapping

After performing the mapping, configure the ServiceNow Adapter.

Figure 9.4 shows an example of the configuration of the Adapter's Connection tab.

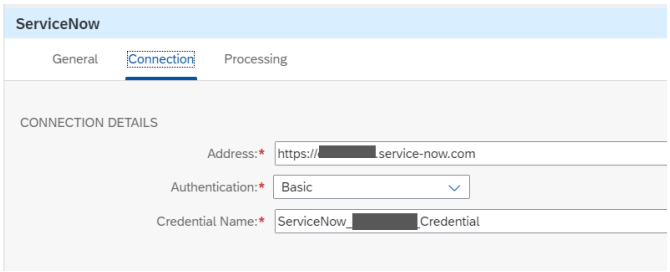

Figure 9.4 Connection tab

In the Connection Tab (shown in Figure 9.4), supply the ServiceNow details. In this example, Basic Authentication is used, and the Credential Name will need to be deployed in Cloud Integration. After configuring the Connection Tab, configure the Processing Tab.
Figure 9.5 shows the necessary configuration details.

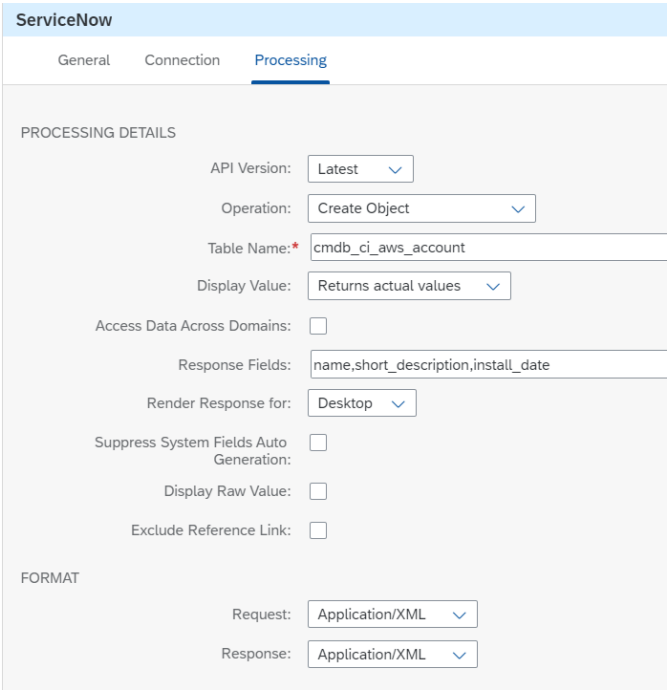

Figure 9.5 Configuration of the Processing tab

In the Processing Tab, select the API Version (which needs to match the version used to generate the XSD in the Eclipse Plug-in).

For this scenario, select **Latest**. Then select **Create Object** as Operation and specify the Table Name as **cmdb_ci_aws_account**.

For the **Response fields**, specify the following fields: **name**, **short_description**, **install_date**.
Use XML as the format for both Request and Response.

Once all necessary configurations are completed, the integration flow can be deployed and then called.

Figure 9.6 shows an example payload created after the mapping that will later be sent to ServiceNow.

```
<request>
    <entry>
        <install_date>2020-07-27T06:38:11Z</install_date>
        <name>Test Account</name>
    </entry>
</request>
```

Figure 9.6 Request XML to ServiceNow

As a result, a new account is created in ServiceNow with the information extracted from Microsoft Dynamics CRM.

10. HEADERS AND HTTP CODES

This section discusses Headers and the HTTP codes applicable to the ServiceNow Adapter.

10.1 Headers

The Adapter allows the sending of Header parameters to ServiceNow. These headers are sent in the form of name/value pairs that can be specified in the Header Parameters section of the Processing Tab. See Figure 10.1.

The table shown below enables the adding of as many Header Parameters as needed by clicking the **Add** button. Fill the lines with a header's name and a value. Note that dynamics values can also be used in the header as seen in Figure 10.1.

Figure 10.1 Create Object - Format & Header Parameters

For instance, a header to enable debugging can be added. In which case, an entry with values similar to what can be seen below must be added:

Name: X-WantSessionDebugMessages

Value: true

Note that to have debugging details returned in the results, it is required to enable Session Debug for SQL queries in ServiceNow. This is achieved by going to Application Navigator to enable System Diagnostics > Session Debug > Debug SQL.

Depending on the operation and the selection made in the Processing Tab, the Response can contain various header parameters. For example, the x-total-count header is returned to indicate the total number of records found for the query. It is mandatory to select the Count checkbox for this parameter to be returned.

10.2 HTTP Response Codes

When exchanging messages, ServiceNow returns different headers. In general, ServiceNow uses standard HTTP status codes. These codes are classified as follows:

1xx: Informational

2xx: Success

3xx: Redirection

4xx: Client Error

Table 7 below lists the known response codes applicable to the ServiceNow Adapter and their descriptions.

Code	Message	Description
200	Success	The call was successful, and the response body was returned.
201	Created	The call to create a record in ServiceNow was successful. The response body was returned.
204	Success	The call was successful. No response body was returned.
400	Bad Request	ServiceNow does not consider the call made as valid. Likewise, the operation failed for an unknown cause.
401	Unauthorized	The user-related credentials used in the Connection tab are not authorized to use the ServiceNow API.
403	Forbidden	The user-related credentials used in the Connection tab are not allowed or permitted to use the operation specified in the Processing tab.
404	Not Found	The resource or object requested does not exist or was not found.
406	Not Acceptable	The endpoint used does not support the format that was requested for the response.
415	Unsupported Media Type	The endpoint used does not support the format that was requested for the request.

Table 7 Status codes and description

11. SUPPORT AND TROUBLESHOOTING

In case of issues or errors, change the Log level of the integration flow to **Traces**. This can be done in SAP Cloud Integration via the Monitor > Manage Integration Content page. An example of such a configuration can be seen in Figure 11.1.

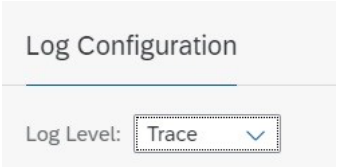

Figure 11.1 Activating Traces

The Trace Log level enables the collection of more traces that can be used to effectively understand the problem. These traces can also be used in a ticket.

The next section discusses a few issues that might be encountered and workable solutions.

11.1 SSL Handshake Exception

Error Message:

javax.net.ssl.SSLHandshakeException: sun.security.validator.ValidatorException: PKIX path building failed: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target, cause: sun.security.provider.certpath.SunCertPathBuilderException: unable to find valid certification path to requested target.

Possible Solution:

This error means that in the Keystore of SAP Cloud Integration, an entry for the root certificate of ServiceNow is missing. At the time this document is published, ServiceNow expects an Entrust Root certificate.

11.2 No Artifact Descriptor Found

Error Message:

[CAMEL][IFLOW][CAUSE]: Cause: com.sap.it.nm.types.NodeManagerException: [CONTENT][CONTENT_DEPLOY][NoArtifactDecriptorFoundForArtifactName]:No artifact descriptor found for artifactName ServiceNowOauth.

Possible Solution:

This error shows that one of the security artifacts used in the Connection tab in the Adapter does not exist. Ensure that the artifact used in the connection tab exists as a Security Material.

11.3 Invalid Table xxxx

Error Message:

com.sap.it.rt.adapter.http.api.exception.HttpResponseException: An internal server error occured: <?xml version="1.0" encoding="UTF-8"?><response><error><detail>null</detail><message>Invalid table incident1</message></error><status>failure</status></response>.

Possible Solution:

This error indicates that the table provided in the Table Name property of the Adapter (in the Processing Tab) does not exist in ServiceNow. In the example error, it states that table *incident1* is correct.

Check and ensure that the Table Name provided in the Processing Tab exists. A list of Table Names can be searched using the Eclipse Workbench.

12. REFERENCES

12.1 Create and OAuth Endpoint for the ServiceNow Adapter

To activate OAuth2 authentication in the ServiceNow Adapter and plug-in, an application needs to be registered in the ServiceNow tenant.

To do so, proceed as follows:

Go to **System OAuth** (or search for it) and select **Application Registry**. Click on **New** as shown in Figure 12.1.

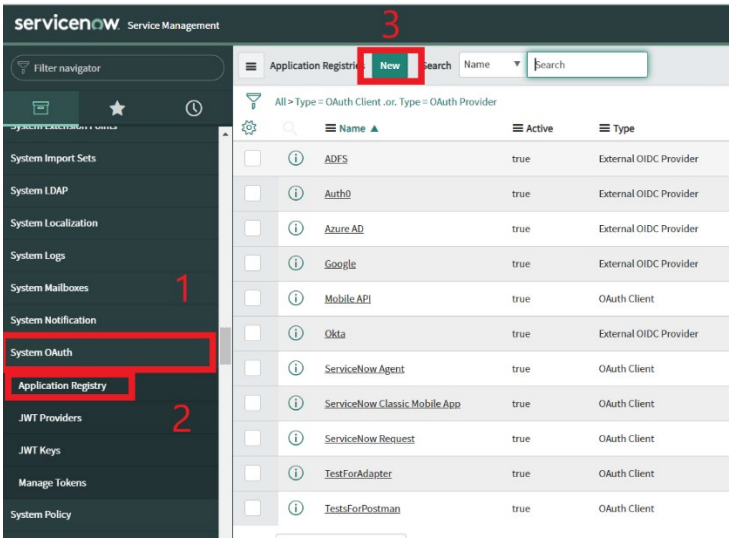

Figure 12.1 Register a new Application for OAuth

Select **Create an OAuth API endpoint for external clients.**
Define a **Name** for the API endpoint and save the **Client ID** for later use. Similarly, define and save a **Client Secret**. Add comments as seen fit.
Click on **Submit**, as shown in Figure 0.2.

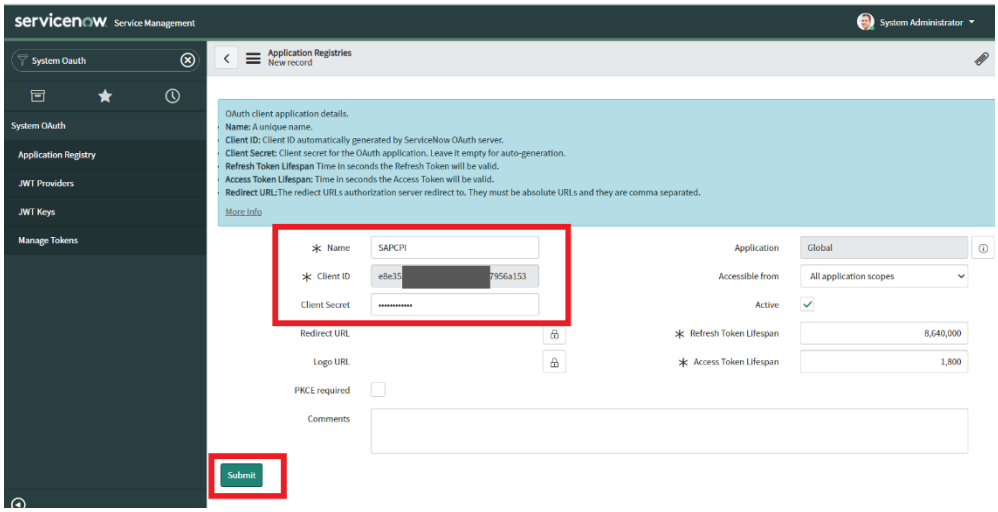

Figure 0.2 Creation of an OAuth API endpoint

The details of the Client ID and Client Secret can be used later in the Adapter.

www.sap.com/contactsap

THE BEST RUN SAP