Oracle® Banking Microservices Architecture

Routing Hub Configuration Cloud Service User Guide

Release 14.7.5.0.0
G17051-02
September 2024

Oracle Banking Microservices Architecture Routing Hub Configuration Cloud Service User Guide, Release 14.7.5.0.0

G17051-02

Contents

Index

Preface

Purpose

This guide enables the user to integrate Oracle Products with External Product Processor through Oracle Banking Routing Hub Platform.

Audience

This guide is intended for the customers and partners.

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

For more information on any related features, refer to the following documents

- *Oracle Banking Common Core User Guide*
- *Oracle Banking Getting Started User Guide*

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.

Screenshot Disclaimer

Personal information used in the interface or documents is dummy and does not exist in the real world. It is only for reference purposes.

Acronyms and Abbreviations

The list of the acronyms and abbreviations used in this guide are as follows:

Table 1 Acronyms and Abbreviations

Abbreviation	Description
API	Application Programming Interface
JSON	Java Script Object Notation
XML	Extensible Markup Language
WSDL	Web Services Description Language

Basic Actions

Table 2 Basic Actions

Action	Description
Submit	Click to complete the transaction after you specify all the input parameters for a particular transaction.
Cancel	Click to cancel the transaction input midway without saving any data.

Table 2 (Cont.) Basic Actions

Action	Description
Clear	Click to clear the transaction input data. The system displays a pop-up screen with confirmation to clear data. You can click **OK** to confirm or click **x** icon to retain the data.
Query	On completion of input of necessary parameters, click this button to fetch and display the details.
OK	Click to confirm the details in the pop-up screen.
Save	Click to save the details specified in the screen.
Exit	Click to close the screen and go to Home screen.

Symbols and Icons

This guide has the following list of symbols and icons.

Table 3 Symbols and Icons - Common

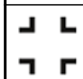
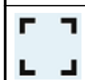
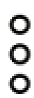
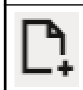
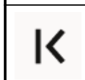
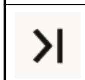
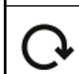
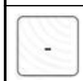
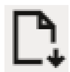
Symbol/Icon	Function
	Minimize
	Maximize
	Close
	Perform Search
	Open a list
	Add a new record
	Navigate to the first record
	Navigate to the last record
	Navigate to the previous record
	Navigate to the next record
	Refresh
	Click this icon to delete a row, which is already added.

Table 3 (Cont.) Symbols and Icons - Common

Symbol/Icon	Function
	Calendar
	Alerts
	Import a file
	Edit a file

1 Introduction

FSGBU Banking Products integrate seamlessly and standardized with Oracle Banking Routing Hub through the use of configurations. The product infrastructure solution includes this component. With Oracle Banking Routing Hub, banking products can be integrated loosely.

Consumer Application The product that requires integration with another product for retrieving information or posting transactions does not need to know the following details while coding.

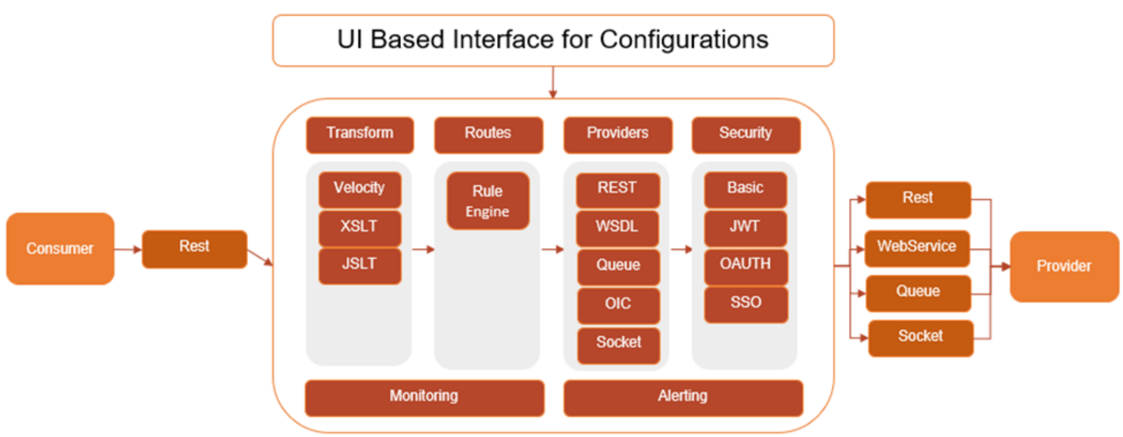
- **Servicing Providers or Product Processors**: The consumer application requests data from the products when required, or a consumer application initiates a transaction for the products to post.
- **Name of the Service**: Logical name of the service example: The service provider's product allows us to fetch details or initiate a transaction for Logical names like Funds Transfer and Letter of Credit.
- **Messaging structure of Service**: Structure of the message example: JSON, XML, multipart request.
- **Communication Protocol**: Web services, Rest API, Queue, OIC, and SOCKET.

Through the 'Oracle Banking Routing Hub', consumers can achieve and modify integration, and they can integrate with different versions of a single product processor if necessary.

This guide shows the maintenance of two product as given below.

- Oracle Service Consumer as Service Consumer
- External Product Processor as Service Provider

Figure 1-1 UI Based Interface for Configurations



2
Service Consumers

This topic describes the systematic instructions to configure the service consumers.

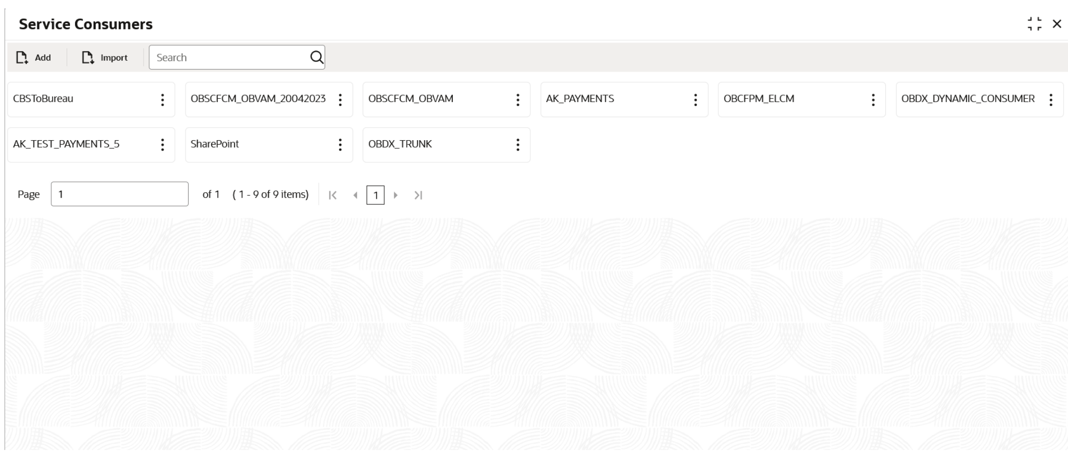
Service Consumer is an Oracle banking product which invokes Oracle Banking Routing Hub API for integration. Oracle Banking Routing Hub analysis, evaluate destination product processor and transform data into format as required by the destination product processor for service a request type

Specify **User ID** and **Password**, and login to **Home** screen.

1. On **Home** screen, click**Core Maintenance**. Under **Core Maintenance**, click **Routing Hub**.
2. Under **Routing Hub**, click **Service Consumers**.

   The **Service Consumers** screen displays.

   Figure 2-1 Service Consumers

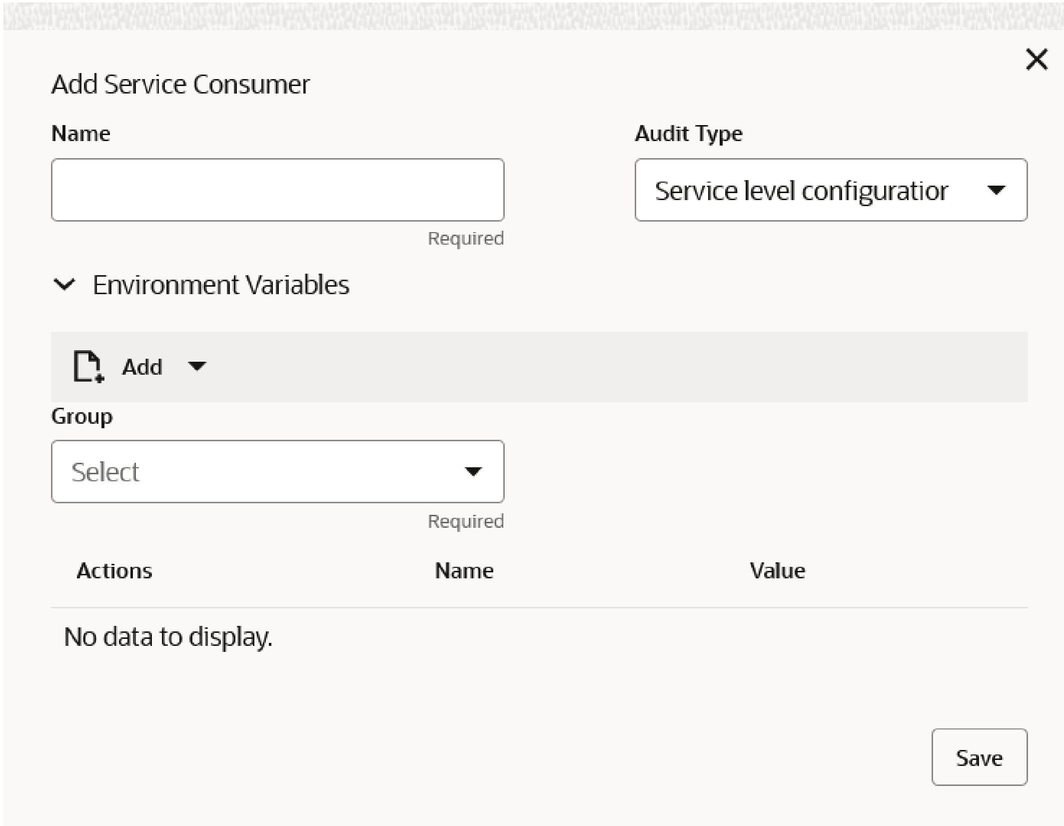
   

Add Service Consumer
In addition to importing Service Consumers, users can create Service Consumers manually using Add option..

3. Click **Add**.

   The **Add Service Consumer** screen displays.

Figure 2-2 Add Service Consumer



4. Specify the fields on **Add Service Consumer** screen.

> **Note:**
>
> The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

Table 2-1 Add Service Consumer - Field Description

Field	Description
Name	Specify the unique service consumer name. **Note:** • Enter 0 to maximum of 255 characters. • No numeric value at beginning and no space allowed.

Table 2-1 (Cont.) Add Service Consumer - Field Description

Field	Description
Audit Type	Select the Audit type from the dropdownlist. The available options are: • **All Requests** - All requests are logged in the OBRH and can be viewed later for debugging. • **Service level configuration** - Option has been provided at consumer services for enabling audit of requests for specific Consumer Services. Audit type should be configured as “Service level configuration” and audit option at “Consumer Services” should be selected for Consumer Services which need to be audited. Monitoring dashboard does not provide the data for requests which are not being audited. • **None** - Disables the audit completely. Audit logs cannot be reviewed later and monitoring dashboard does not provide the data
Add	To add, refer to step 5. Select the group from the drop-down list. The available options are: • **Group** • **Variable**
Group	Select the group from the drop-down list.
Action	Displays the action. The user can edit or delete the header.
Name	Displays the name of the header.
Value	Displays the value of the header.

Environment Variables
The user must define the group of variables which can be accessed throughout the specific consumer’s configuration. The syntax for accessing environment variables is below: $env.Environment_Group_Name.Environment_Variable_Name

For example, $env.COMMON.BRANCH_CODE

5. To add **Environment Variables**, follow the below steps.
   a. Click **Add** on the **Add Service Consumers** screen, and select **Group** from drop-down list to add the group.

      The **Add Environment Group** screen displays.

Figure 2-3 Add Environment Group

b. Specify the fields on **Add Environment Group** screen and click **OK**.

> **Note:**
>
> The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

Table 2-2 Add Environment Group - Field Description

Field	Description
Name	Specify the name of the environment group. **Note:** • Enter 0 to maximum of 255 characters. • No numeric value at beginning and no space allowed.

c. Click **Add** on **Add Service Consumer** screen and select **Variable** from drop-down list to add the variable.

The **Add Environment Variable** screen displays.

Figure 2-4 Add Environment Variable

d. Specify the fields on **Add Environment Variable** screen and click **OK**.

> **Note:**
>
> The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

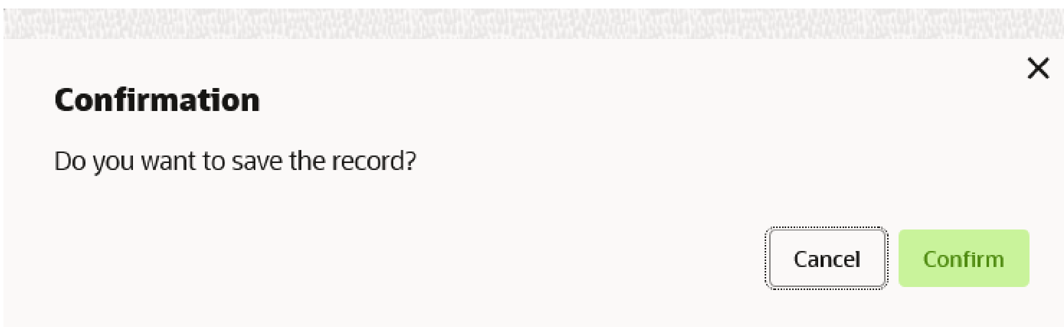
Table 2-3   Add Environment Variable - Field Description

Field	Description
Name	Specify the name of the environment variable. **Note:** • Enter 0 to maximum of 255 characters. • No numeric value at beginning and no space allowed.
Value	Specify the value of the environment variable. The value can either be hardcoded or Velocity mapping.

6. Click **Save** to save the details.

   The **Confirmation** screen displays.

Figure 2-5   Confirmation - Add Service Consumers



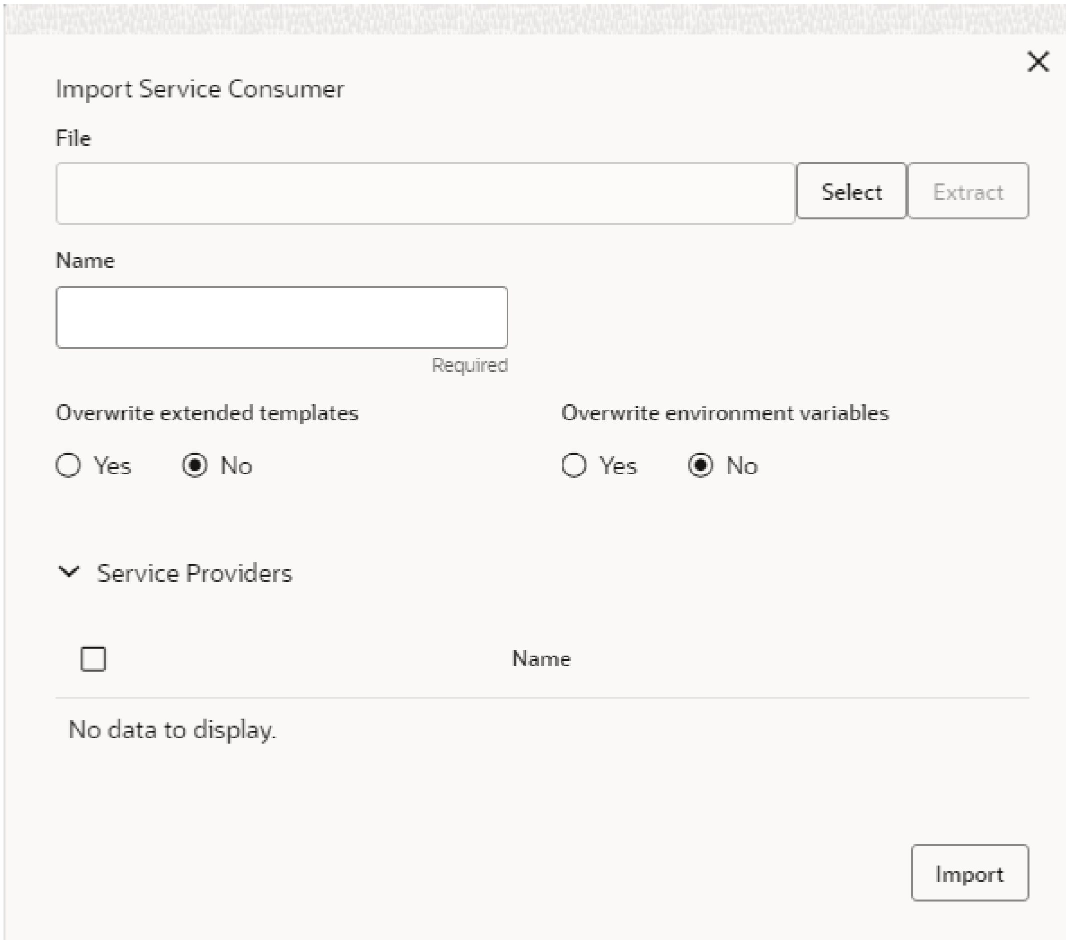
7. Click **Confirm** to save the record.

Import Service Consumer
The user can create a service consumer by importing the JSON file and manually selecting the service providers or select all providers that needs to be imported. The user can also import zip file in order to import all the configuration JSON files together.

8. Click **Import**.

   The **Import Service Consumer** screen displays.

Figure 2-6   Import Service Consumer



9. Specify the fields on **Import Service Consumer** screen.

> **Note:**
>
> The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

Table 2-4   Import Service Consumer - Field Description

Field	Description
File	Select the file using **Select**. **Note:** Allows only to select one file and accepts JSON and ZIP file.

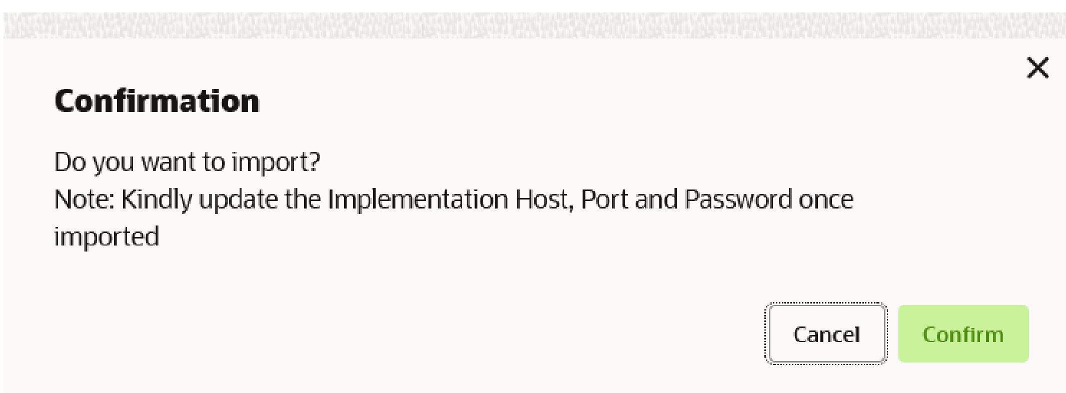
Table 2-4 (Cont.) Import Service Consumer - Field Description

Field	Description
Extract	Extracts the consumer name and service provider list from JSON file only and displays it in the respective elements.
Name	Specify the name of the service provider. **Note:** • Name cannot be blank and required only for JSON file. • Enter 0 to maximum of 255 characters. • No numeric value at beginning and no space allowed.
Overwrite extended templates	Select the respective radio button to overwrite the extended templates. The available options are: • **Yes** - This option overwrites the extended templates. • **No** - This option retains the existing extended templates.
Overwrite environment variables	Select the respective radio button to indicate whether environment variables (JSON file) should overwrite existing environment variables or not The available options are: • **Yes** - This option overwrites the environment variables. • **No** - This option retains the existing environment variables.
Service Providers	Displays the service provider details.
Name	Displays the list of service providers names that are present in JSON file only.

10. Click **Import** to import the selected service consumer file.

    The **Confirmation** screen displays.

Figure 2-7 Confirmation - Import Service Consumer



11. Click **Confirm** to import the file.

> **Note:**
>
> Below data needs to be changed after importing provider configuration file:
>
> - Implementation Host and Port
> - Implementation Authentication Password

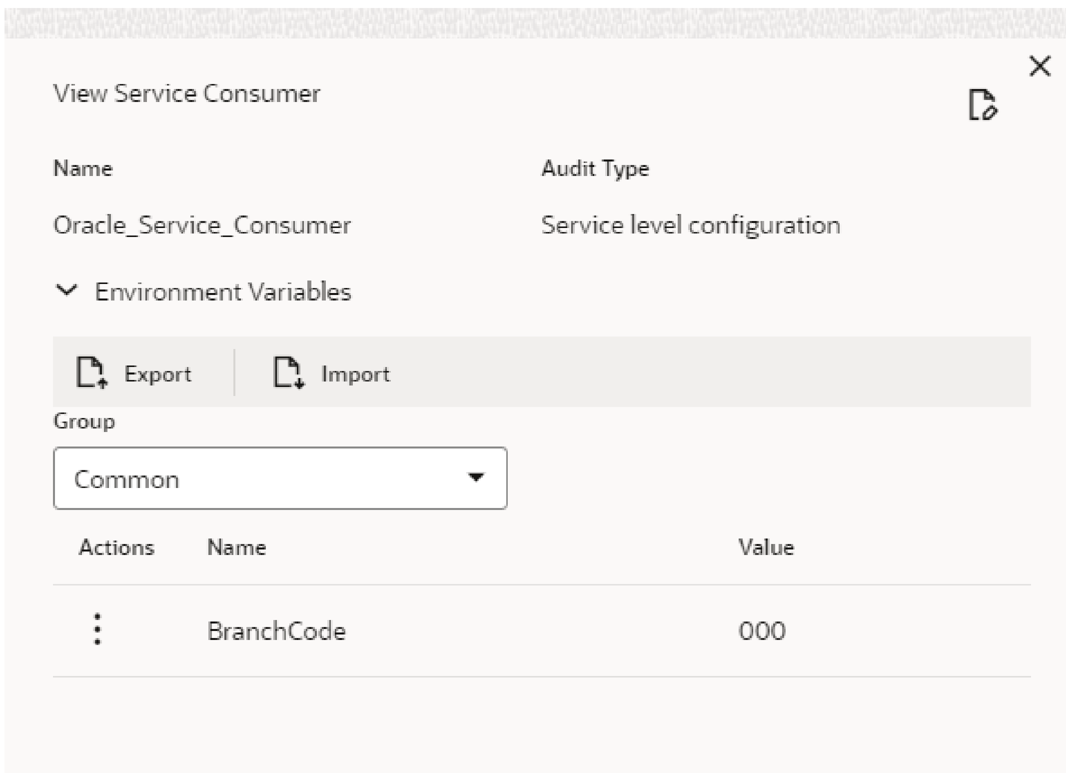
View Service Consumer
The user can view consumer details and can also switch to edit form by clicking on edit icon.

The user can also export and import only for the environment variables.

12. On the **Service Consumer** tile, click , and select **View** from the dropdown list.

    The **View Service Consumer** screen displays.

    Figure 2-8 View Service Consumer



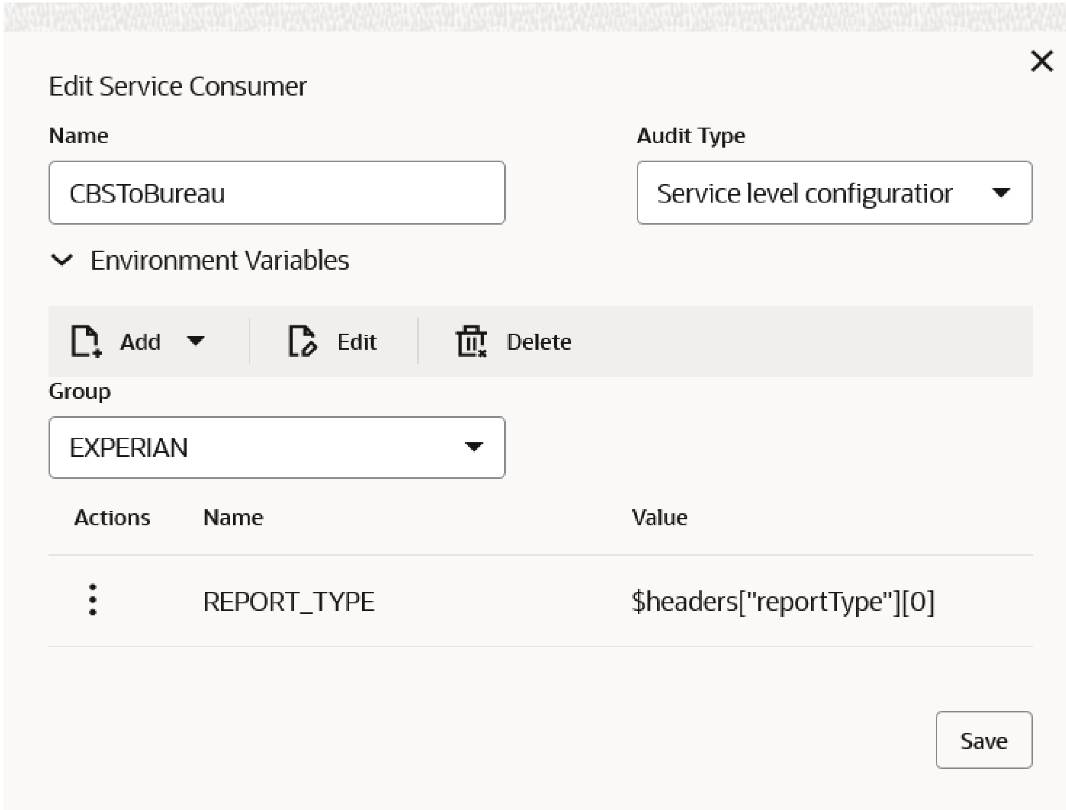
13. Click **Edit** icon to edit the Service Consumer.

Edit Service Consumer
The user can modify the consumer details.

14. On the **Service Consumer** tile, click **Operation Menu** (3 dot icon), and select **Edit** from the dropdown list.

    The **Edit Service Consumer** screen displays.

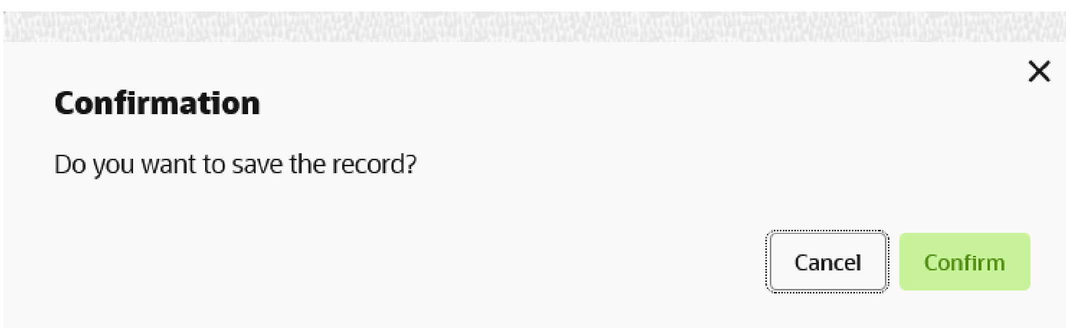
Figure 2-9 Edit Service Consumer



15. Click **Save** to save the modified consumer details.

    The **Confirmation** screen displays.

Figure 2-10 Confirmation - Edit Service Consumer



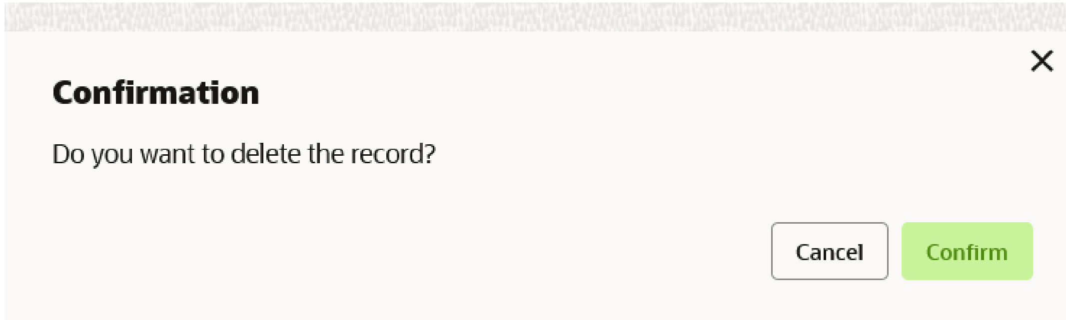
16. Click **Confirm**.

Delete Service Consumer
The user can delete the Service Consumer.

17. On the **Service Consumer** tile, click **Operation Menu** (3 dot icon), and select **Delete** from the dropdown list.

    The **Confirmation** screen displays.

Figure 2-11   Confirmation - Delete



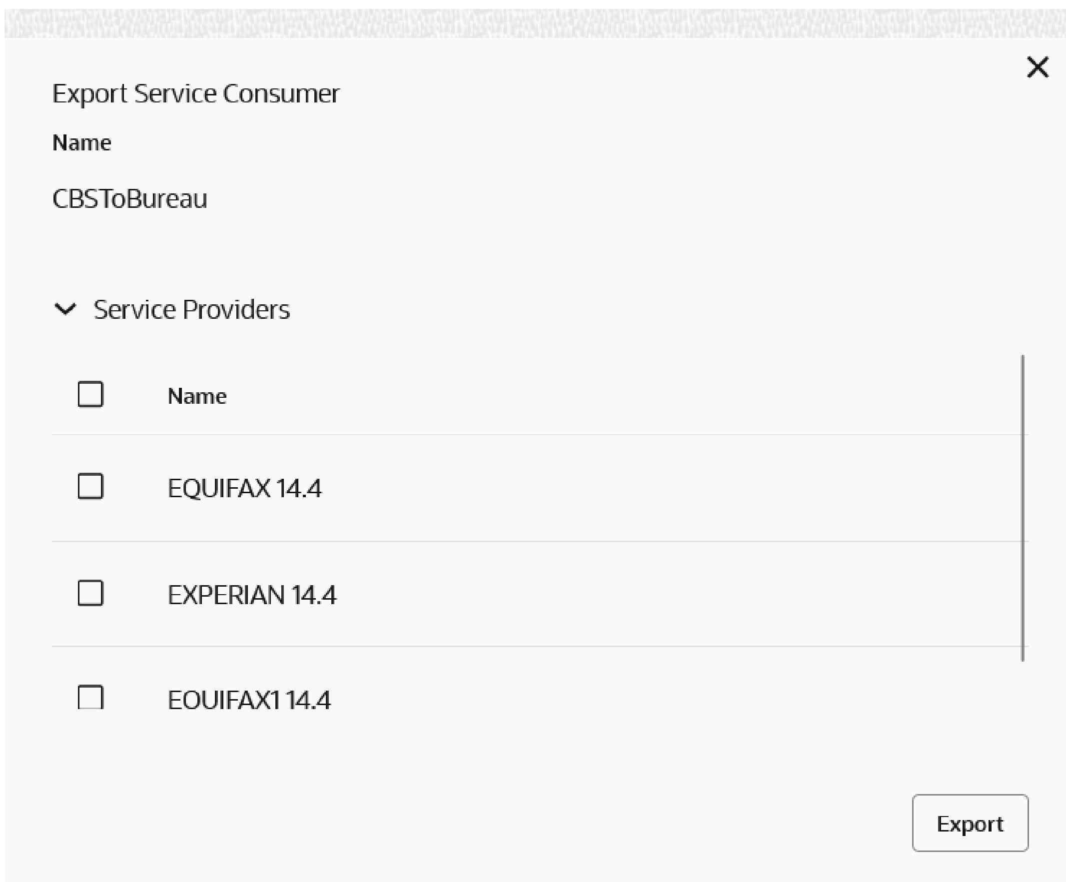
18. Click **Confirm** to delete the service consumer.

JSON Export
User can export the consumer configuration as JSON file. The option for Export is provided to move the configurations from one environment to another.

19. On **Service Consumer** tile, click **Operation Menu** (3 dot icon).
20. On **Export** option, select **JSON** from the list.

    The **Export Service Consumer** screen displays.

Figure 2-12   Export Service Consumer

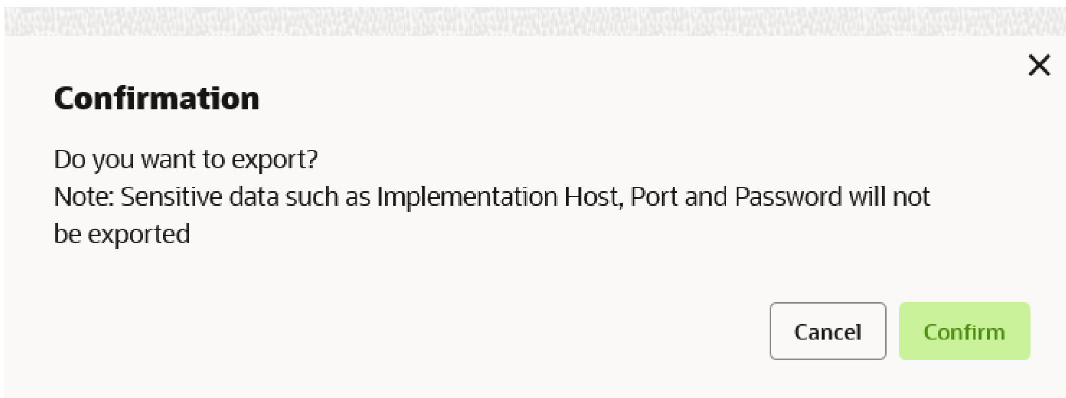


> **Note:**
>
> - The user has an option to select the service providers from the list which needs to be exported or can click on select all for all service providers.
> - The JSON Export feature exports below data:
>   - Selected service consumer
>   - All consumer services
>   - Selected service providers with services
>   - All implementations of selected service providers with services (without Host, Port and Authentication Password)
>   - All transformations
>   - All routes

21. Select the required service providers and click **Export**.

    The **Confirmation** screen displays.

Figure 2-13 Confirmation - JSON Export



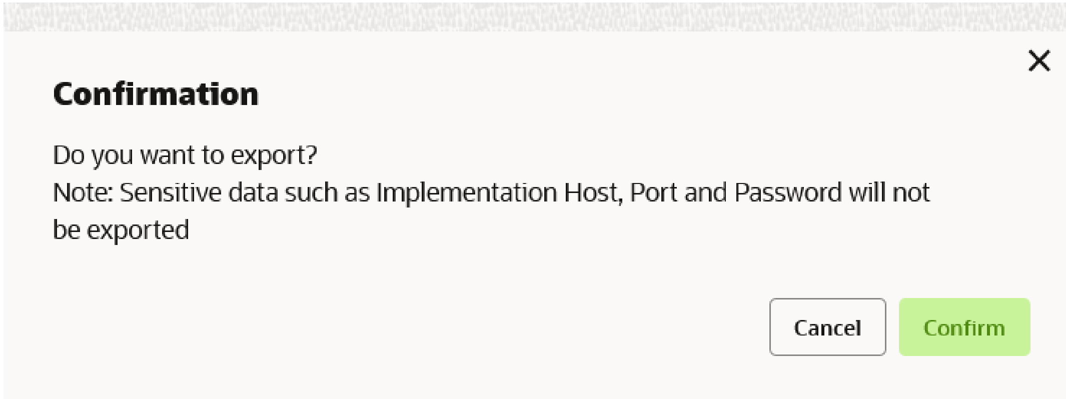
22. Click **Confirm** to export the service consumer in JSON file.

SQL Export
The user can export the consumer configuration as SQL file.

23. On **Service Consumer** tile, click **Operation Menu** (3 dot icon).
24. On **Export** option, select **SQL** from the list.

    The **Confirmation** screen displays.

Figure 2-14   Confirmation



> **Note:**
>
> The SQL Export feature exports entire configuration without Host, Port, and Authentication Password details.

25. Click **Confirm** to export the consumer configuration as SQL file.

Configuration

26. On **Service Consumer** tile, click **Operation Menu** (3 dot icon), and click **Configuration**.

    The **Configuration** screen displays.

> **Note:**
>
> Refer to **Configuration** topic for the screen and field description.

Request Audit

27. On **Service Consumer** tile, click **Operation Menu** (3 dot icon), and click **Request Audit**.

    The **Request Audit** screen displays.

> **Note:**
>
> Refer to **Request Audit** topic for the screen and field description.

3 Service Providers

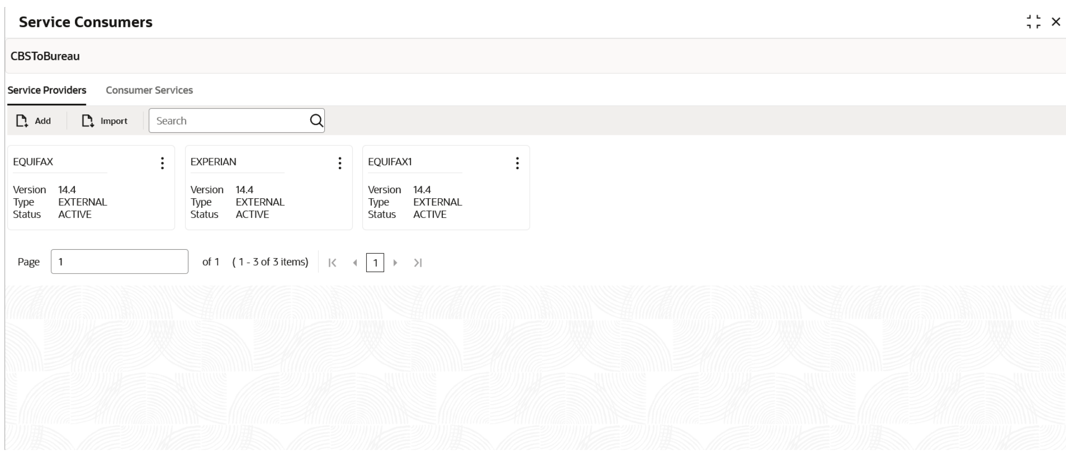
This topic describes the systematic instructions to configure the service providers.

The **Service Providers** are the product processors configure to process request send by Oracle Banking Routing Hub on behalf of service consumers. It comprises destination integration details.

1. On **Service Consumers** screen, click the required service consumer.

   The **Service Providers** screen displays.

Figure 3-1 Service Providers

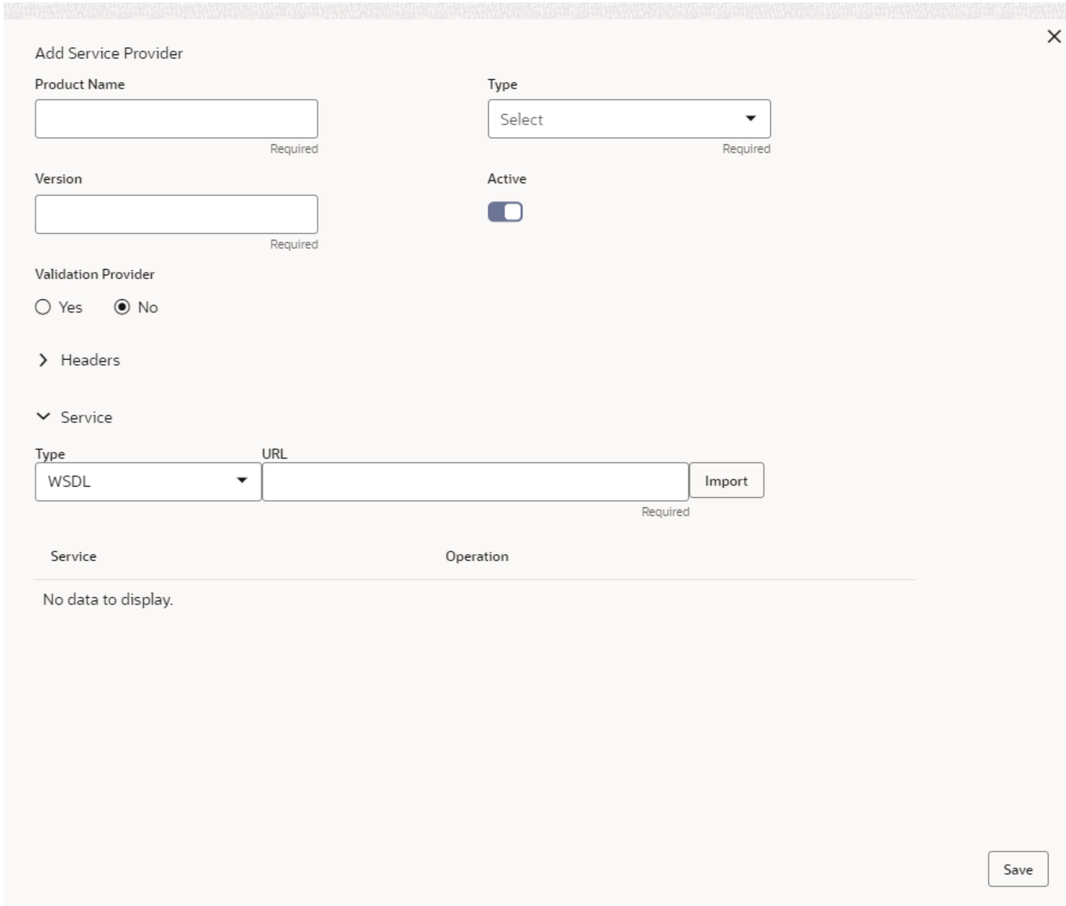


Add Service Provider
The user can create Service Provider manually.

2. Click **Add**.

   The **Add Service Provider** screen displays.

Figure 3-2   Add Service Provider



3. Specify the fields on **Add Service Provider** screen.

> **Note:**
>
> The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

Table 3-1   Add Service Provider - Field Description

Field	Description
Product Name	Specify the product name of the service provider. **Note:** • Enter 0 to maximum of 255 characters. • No numeric value at beginning and no space allowed.

Table 3-1 (Cont.) Add Service Provider - Field Description

Field	Description
Type	Select the type of service provider from drop-down list The available options are: • **INTERNAL**: Used for Oracle products • **EXTERNAL**: Used for non-Oracle products
Version	Specify the provider version. **Note:** • Enter 0 to maximum of 255 characters. • Only numeric or decimal values are allowed.
Active	Predefined values are Active / Inactive If provider is marked as inactive, then all related routes will be stopped.
Validation Provider	Predefined values are Yes / No This property can be enabled to use a separate provider service for validating the requests before sending it for processing.
Add	To add, refer to the below steps.
Actions	Displays the action. The user can edit or delete the header.
Name	Displays the name of the header.
Value	Displays the value of the header.
Type	Select the type of service from drop-down list. The available options are: • **WSDL** • **SWAGGER** • **OTHERS**
Name	Specify the name of the operation. **Note:** This field appears only if the **Type** is selected as **OTHERS**.

Table 3-1 (Cont.) Add Service Provider - Field Description

Field	Description
Http Method	Select the HTTP method. The available options are: • **GET** • **POST** • **PUT** • **PATCH** • **DELETE** **Note:** This field appears only if the **Type** is selected as **OTHERS**.
Endpoint	Specify the endpoint URL for the operation. **Note:** This field appears only if the **Type** is selected as **OTHERS**.
URL	Specify the service URL of the file location. **Note:** This field appears only if the **Type** is selected as **WSDL** and **SWAGGER**.
Import	Click **Import** to extract the service information from URL. **Note:** This field appears only if the **Type** is selected as **WSDL** and **SWAGGER**.
Context Path	Context path of below formatted URL http://host:port/context-path/endpoint

Table 3-1 (Cont.) Add Service Provider - Field Description

Field	Description
Service Headers	Specify the Endpoint specific headers. Value can either be hardcoded or can be Velocity mapping. **Note:** This field appears only if the **Type** is selected as **OTHERS**.
Service Query Params	Specify the Endpoint specific query parameters. Value can either be hardcoded or can be Velocity mapping. **Note:** This field appears only if the **Type** is selected as **OTHERS**.
Service	Displays the extracted service from the selected URL.
Operation	Displays the extracted operation from the selected URL.

Headers
A product processor might require some standard headers to be passed along with the request. The user can specify the headers which are required by service endpoints for its all implementations but not present in swagger file.

> **Note:**
>
> Content-type header will be removed from Provider request if header value is NONE.

4. To add **Headers**, follow the below steps.
   a. Under **Headers** section, click **Add**.

      The **Add Header** screen displays.

Figure 3-3   Add Header

b. Specify the fields on **Add Header** screen.

> **Note:**
>
> The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

Table 3-2   Add Header - Field Description

Field	Description
Name	Specify the name of the header.
Value	Specify the value of the header.

5. Click **OK** to save the details.

   The **Confirmation** screen displays.

6. Click **Confirm**.

Service

- **WSDL**:
  The Web Services Description Language (WSDL) is an XML-based interface description language that is used for describing the functionality offered by a web service.

  Both SSL and non-SSL WSDL URL are supported.

  > **Note:**
  >
  > If there is a change in wsdl file, then same wsdl file need to be imported again to update the provided service information in Routing Hub.

- **SWAGGER**:
  Swagger is an Interface Description Language for describing RESTful APIs expressed using JSON.

  Currently, Swagger 2.0 & OpenAPI 3.0 both are supported.

  > **Note:**
  >
  > If there is a change in swagger file, then same swagger file need to be imported again in order to update the provided service information in Routing Hub.

- **OTHERS**:
  OTHERS option is selected for adding REST API details manually when provider does not have swagger file.

  > **Note:**
  >
  > If there is a change in existing endpoint, then the same endpoint details need to be entered again with the new changes in order to update the existing provided service information in Routing Hub.

Import Service Provider
The user can create a service provider by importing the JSON file and also can import zip file in order to import all the configuration JSON files together (except parent level configuration JSON files).

7. Click **Import**.

   The **Import Service Provider** screen displays.

   Figure 3-4 Import Service Provider

For more information on fields, refer to the field description table below.

Table 3-3 Import Service Provider - Field Description

Field	Description
File	Select the file using **Select** button. **Note:** Allows only to select one file and accepts JSON and ZIP file.
Overwrite extended templates	Select the respective radio button to overwrite extended templates. The options are: • **Yes** - This option overwrites the extended templates in configuration. • **No** - This option retains the existing extended templates in configuration. **Note:** This field appears only if the ZIP File is selected.

8. Click **Import** to import the selected file.

   The **Confirmation** screen displays.

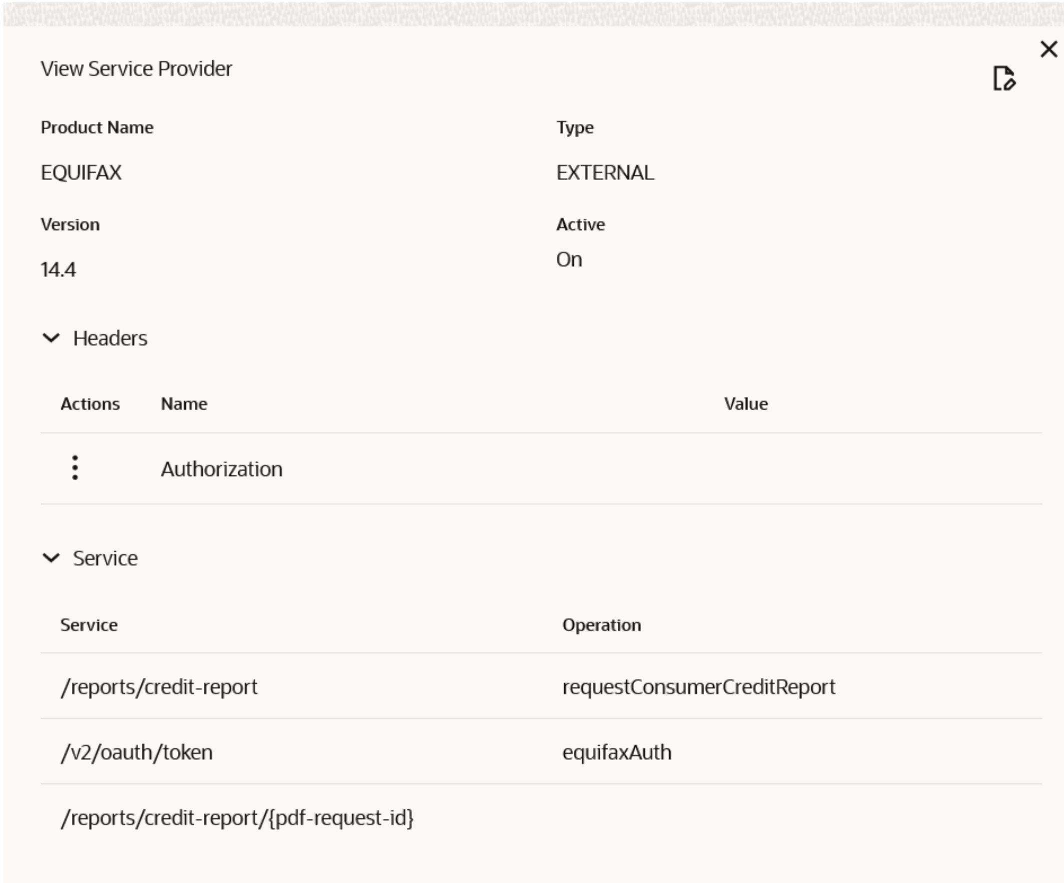
> **Note:**
>
> The following data needs to be changed after importing provider configuration file:
>
> - Implementation Host and Port
> - Implementation Authentication Password

View Service Provider

9. On **Service Provider** tile, click **Operation menu** (3 dots button), and click **View**.

   The **View Service Provider** screen displays.

Figure 3-5 View Service Provider



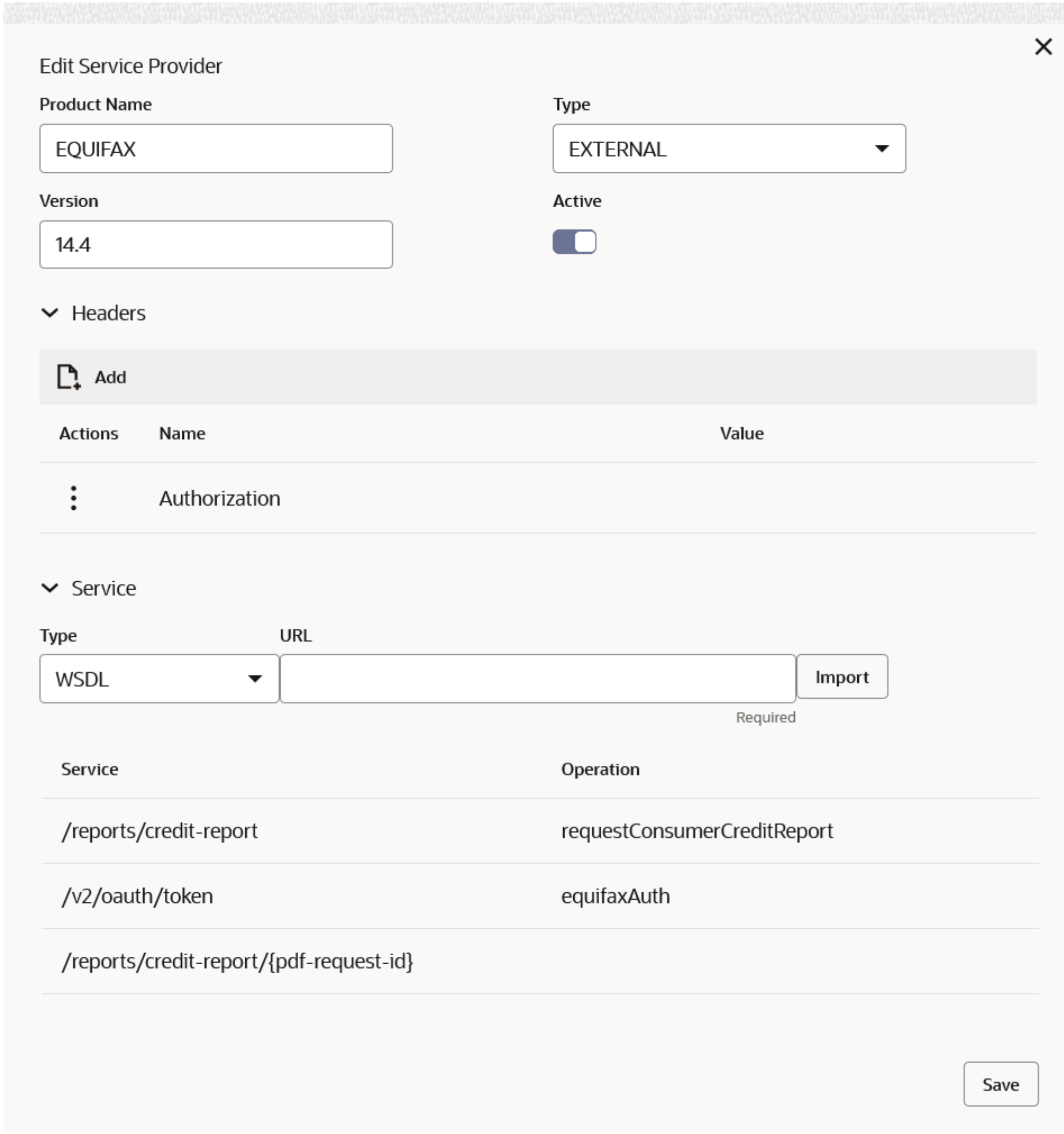
10. Click **Edit** button to edit the Service Provider.

Edit Service Provider
The user can modify the provider details.

11. On **Service Provider** tile, click **Operation menu** (3 dots button), and click **Edit**.

    The **Edit Service Provider** screen displays.

Figure 3-6 Edit Service Provider



12. Click **Save** once the edit is done.

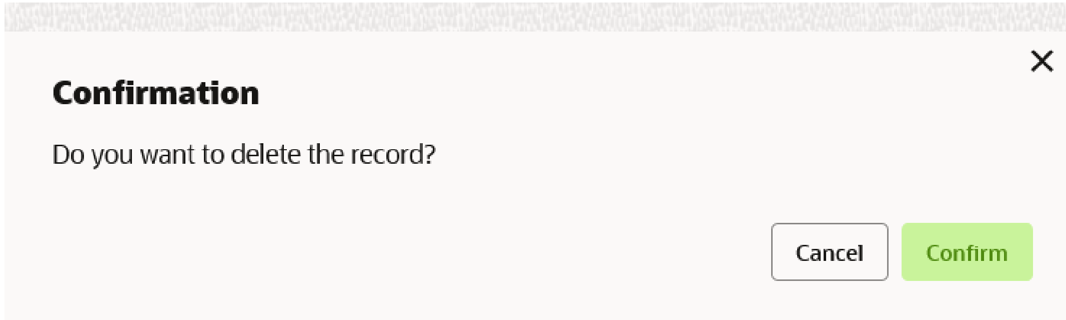
    The **Confirmation** screen displays.

Delete Service Provider
The user can delete the provider.

13. On **Service Provider** tile, click **Operation menu** (3 dots button), and click **Delete**.

    The **Confirmation** screen displays.

Figure 3-7 Confirmation - Delete



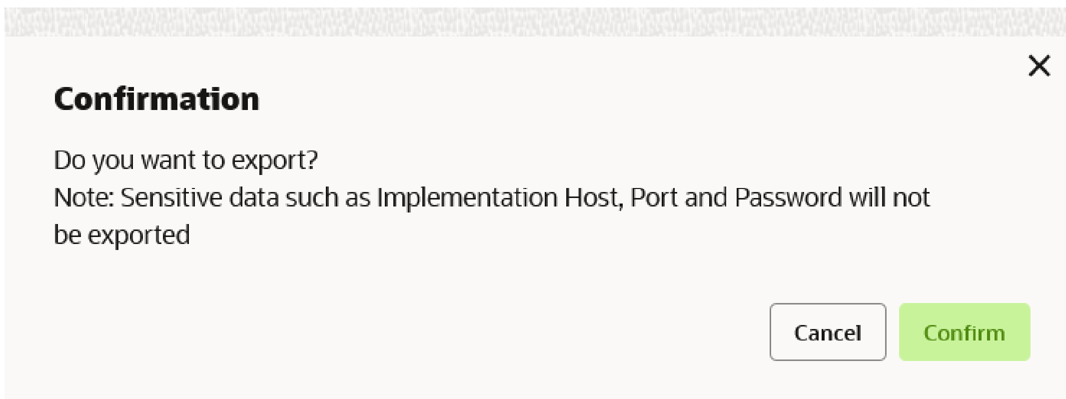
14. Click **Confirm** to delete the selected Service Provider.

Export Service Provider
The user can export the provider configuration as JSON file.

15. On **Service Provider** tile, click **Operation menu** (3 dots button), and click **Export**.

    The **Confirmation** screen displays.

Figure 3-8 Confirmation - Export



> **Note:**
>
> The below data cannot be exported:
>
> - Implementation Host
> - Implementation Port
> - Implementation Authentication Password
>
> The above data needs to be configured manually after importing the configuration file. Same has been mentioned in Import section.

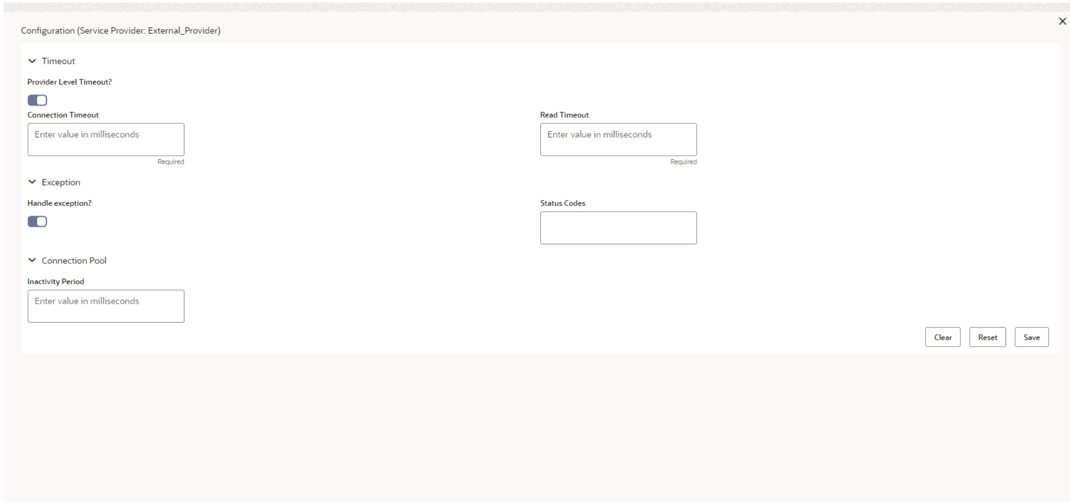
16. Click **Confirm** to export the selected Service Provider.

Configuration
End-user can configure the properties for failing the routing hub requests.

17. On **Service Provider** tile, click **Operation menu** (3 dots button), and click **Configuration**.

    The **Configuration** screen displays.

Figure 3-9 Configuration



For more information on fields, refer to the field description table below

Table 3-4 Configuration Service Provider - Field Description

Field	Description
Provider level timeout	This property is used to override the global timeout values. **Note:** Default value is false.
Connection Timeout	This property is used to set the timeout in making the initial connection i.e. connection handshake. **Note:** Value should be in milliseconds.
Read Timeout	This property is used to set the timeout on waiting to read data. **Note:** Value should be in milliseconds.

Table 3-4 (Cont.) Configuration Service Provider - Field Description

Field	Description
Handle exception	This property is used to fail the routing hub request for failed provider requests. **Note:** Default value is false.
Status Codes	This property is used to fail routing hub request for specific status codes of failed provider requests. If not specified, then routing hub request will fail for all 4xx and 5xx status codes of failed provider requests.
Inactivity Period	This property is used to specify connection inactivity time for re-validating connections in connection pool.

Request Audit

18. On **Service Provider** tile, click **Operation menu** (3 dots button), and click **Request Audit**.

    The **Request Audit** screen displays.

> **Note:**
>
> Refer to **Request Audit** topic for the screen and field description.

Clear Cache

The user can clear the SOAP client cache for the service providers.

19. On **Service Provider** tile, click **Operation menu** (3 dots button), and click **Clear Cache**.

4 Implementation

This topic provides the systematic instructions to configure the implementation.

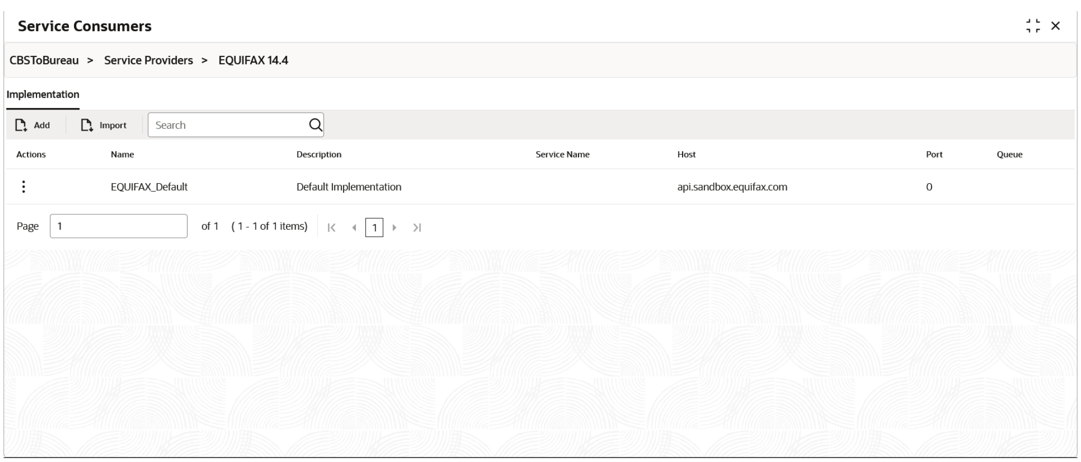
The implementation comprises of Eureka client instance, Host, Port, authentication, and implementation specific service details. Oracle Banking Routing Hub supports web services and Rest API.

> **Note:**
>
> Default implementation is created whenever a new service provider is added.

1. On **Service Provider** screen, click on the required service provider tile.

   The **Implementation** screen displays.

   Figure 4-1 Implementation

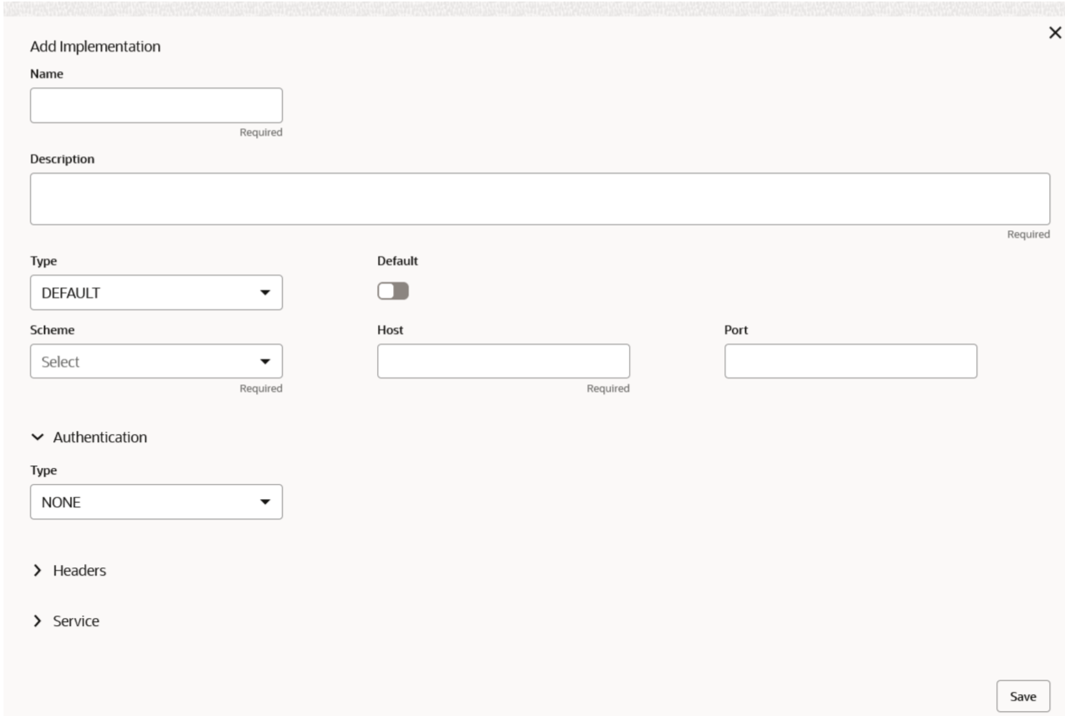
   

Add Implementation
The user can create the implementation manually.

2. Click **Add**.

   The **Add Implementation** screen displays.

Figure 4-2 Add Implementation



3. Specify the fields on **Add Implementation** screen.

> **Note:**
>
> The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

Table 4-1 Add Implementation - Field Description

Field	Description
Name	Specify the name of the implementation. **Note:** • Enter 0 to maximum of 255 characters. • No numeric value at beginning and no space allowed.

Table 4-1 (Cont.) Add Implementation - Field Description

<table>
<tr><th>Field</th><th>Description</th></tr>
<tr><td>Description</td><td>Specify the description of the implementation.

Note:
• Enter 0 to 1000 characters.
• No space allowed at beginning or ending of the characters.</td></tr>
<tr><td>Type</td><td>Select the type of implementation from drop-down list.
The available options are:
• DEFAULT
• QUEUE
• OIC
• SOCKET
DEFAULT type is for REST and SOAP API calls.

Note:
The type as OIC is only applicable for cloud services.</td></tr>
<tr><td>Default</td><td>Toggle the button if user wants to default.
Each type can have one default implementation.</td></tr>
<tr><td>Single Tenant</td><td>Select the toggle to append tenant details with eureka VIP for services which are registered on eureka as single tenant services.

Note:
This field is available only for internal providers and applicable only for Cloud.</td></tr>
</table>

Table 4-1 (Cont.) Add Implementation - Field Description

Field	Description
Eureka Instance	**Eureka Instance** is available only for internal providers and default type. By default, **Eureka Instance** will be toggled ON for internal providers and OFF for external providers. **Note:** If the Eureka Instance is toggled ON, the Api-gateway will be removed (if present) from the request URL sent to the provider. If the Eureka Instance is toggled OFF and the authentication type is selected as JWT_TOKEN or OAuth_TOKEN, the provider request URL will include apigateway if it's missing. If the Eureka Instance is activated, it propagates the userId, branchCode, piienabled, languageCode, and locale headers from the routing hub request to the service provider request.
Scheme	Select the scheme from drop-down list The available options are: • **http** • **https** **Scheme** option is available only for default type.
Service Name	If **Eureka Instance** is toggled ON and type is default, then only service name is required.
Host	Specify the host. **Note:** • Host cannot be blank. • Enter 0 to 255 characters. • Space is not allowed. If **Eureka Instance** is toggled OFF and type is default, then only host and port is required.
Port	Specify the port number. **Note:** • Enter 0 to 6 characters. • Enter only numeric value. If **Eureka Instance** is toggled OFF and type is default, then only host and port is required.
Authentication	The below fields appear only if **Eureka Instance** is toggled **OFF** and Implementation **Type** is selected as **Default**.

Table 4-1 (Cont.) Add Implementation - Field Description

Field	Description
Type	Select the type of authentication from drop-down list. The available options are: • **BASIC** • **JWT_TOKEN** • **OAUTH_TOKEN** • **SSO**
Encryption	Select the toggle to encrypt user credentials. **Note:** This field is applicable only for **JWT_TOKEN** and **OAUTH_TOKEN** types. **Note:** This field depends on the value of api-gateway's property "EncryptionFlag" at provider end. For more information on property value, please refer to the **Oracle Banking Microservices Architecture Deployments** section in **Oracle Banking Microservices Platform Foundation Installation Guide**.
Username	Specify the name of the user. **Note:** • Enter 0 to maximum of 255 characters. • No numeric value at beginning and no space allowed.
Password	Specify the password.
Headers	The below fields appear only if the Implementation **Type** is selected as **Default**or **OIC**.
Add	Click this button to add header.
Actions	Displays the action. The user can edit or delete the header.
Name	Displays the name of the header.
Value	Displays the value of the header.
Service	The below fields appear only if the Implementation **Type** is selected as **Default** or **OIC**.
Type	Select the type of service from drop-down list. The available options are: • **WSDL** • **SWAGGER** • **OTHERS**

Table 4-1 (Cont.) Add Implementation - Field Description

Field	Description
URL	Specify the service URL of the file location. **Note:** This field appears only if the **Type** is selected as **WSDL** and **SWAGGER**.
Name	Specify the name of the operation. **Note:** This field appears only if the **Type** is selected as **OTHERS**.
Http Method	Select the HTTP method. The available options are: • **GET** • **POST** • **PUT** • **PATCH** • **DELETE** **Note:** This field appears only if the **Type** is selected as **OTHERS**.
Endpoint	Specify the endpoint URL for the operation. **Note:** This field appears only if the **Type** is selected as **OTHERS**.
Content path Prefix	Context path of below formatted URL. http://host:port/context-path/endpointGateway

Table 4-1 (Cont.) Add Implementation - Field Description

Field	Description
Import	Click **Import** to extract the service information from URL and displays it in the Service list. **Note:** This field appears only if the **Type** is selected as **WSDL** and **SWAGGER**.
Service Headers	Specify the Endpoint specific headers. Value can either be hardcoded or can be Velocity mapping. **Note:** This field appears only if the **Type** is selected as **OTHERS**.
Service Query Params	Specify the Endpoint specific query parameters. Value can either be hardcoded or can be Velocity mapping. **Note:** This field appears only if the **Type** is selected as **OTHERS**.
Add	Click this button to add the endpoint details in the Service list.
Service	Displays the extracted service from the selected URL.
Operation	Displays the extracted operation from the selected URL.

Authentication:
If External Product processor require authentication to connect to it, Oracle Banking Routing Hub provides standard authentication mechanism schemes like BASIC, JWT, OAUTH_TOKEN, SSO.

> **Note:**
>
> In case of no authentication, NONE needs to be set as Authentication Type.
> In case of identity propagation, SSO needs to be set as Authentication Type.
> Token is cached for JWT, OAUTH_TOKEN authentication type and OIC implementation type

Services

- **WSDL**:

The Web Services Description Language (WSDL) is an XML-based interface description language that is used for describing the functionality offered by a web service.

Both SSL and non-SSL WSDL URL are supported.

> **Note:**
>
> If there is a change in wsdl file, then same wsdl file need to be imported again to update the provided service information in Routing Hub.

- **SWAGGER**:
  Swagger is an Interface Description Language for describing RESTful APIs expressed using JSON.

  Currently, Swagger 2.0 & OpenAPI 3.0 both are supported.

  > **Note:**
  >
  > If there is a change in swagger file, then same swagger file need to be imported again in order to update the provided service information in Routing Hub.

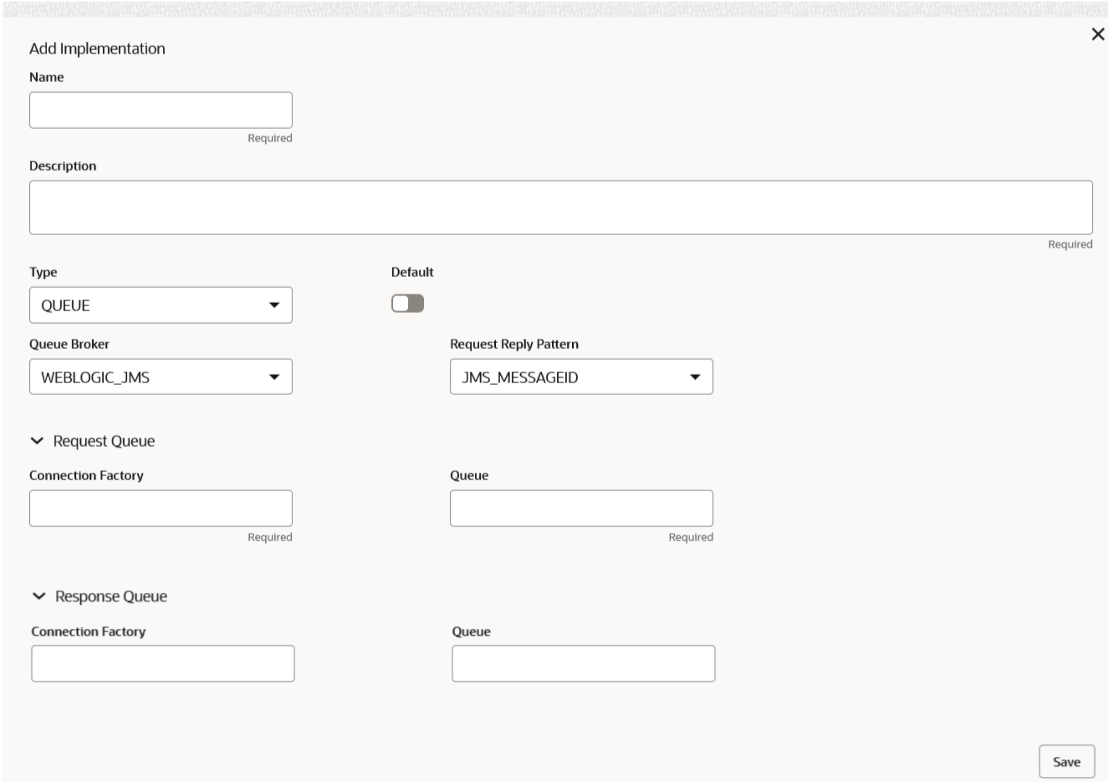
- **OTHERS**:
  OTHERS option is selected for adding REST API details manually when provider does not have swagger file.

  > **Note:**
  >
  > If there is a change in existing endpoint, then the same endpoint details need to be entered again with the new changes in order to update the existing provided service information in Routing Hub.

Queue

If the Implementation **Type** is selected as **Queue**,

Figure 4-3   Add Implementation - Queue



For **QUEUE** type, refer to the field description table below.

Table 4-2   Add Implementation - Queue - Field Description

Field	Description
Type	Select the type of implementation from drop-down list The available options are: • **DEFAULT** • **QUEUE** **DEFAULT** type is for REST and SOAP API calls.
Default	Select the toggle if the user wants to default.
Queue Broker	Select the queue broker from drop-down list. The available options are: • **WEBLOGIC_JMS**
Request Reply Platform	Select the queue broker from drop-down list. The available options are: • **JMS_MESSAGEID** • **JMS_CORRELATIONID** **JMS_MESSAGEID** is default request-reply pattern.
Connection Factory	Specify the connection factory. Connection Factory is JNDI based connection factory name which is used to create connection for JMS client.
Queue	Specify the queue. Queue Name is JNDI based destination name.

Table 4-2 (Cont.) Add Implementation - Queue - Field Description

Field	Description
Connection Factory	Specify the connection factory. Response Connection Factory is needed when destination is going to respond back after processing the request.
Queue	Specify the queue. Response Queue Name is needed when destination is going to respond back after processing the request.

Headers:
External product processor might require some standard headers to be passed along with the request. User can specify the headers which are required by service endpoints for its all implementations but not present in swagger file.

> **Note:**
>
> Content-type header will be removed from Provider request if header value is NONE.

4. Follow the below steps to add **Headers**.
   a. Click **Add** under **Header** section.

      The **Add Header** screen displays.

      Figure 4-4 Add Header

   b. Specify the fields on **Add Header** screen and click **OK**.

      > **Note:**
      >
      > The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

Table 4-3 Add Header - Field Description

Field	Description
Name	Specify the name for the header.
Value	Specify the value for the header.

5. Click **OK** to save the details.

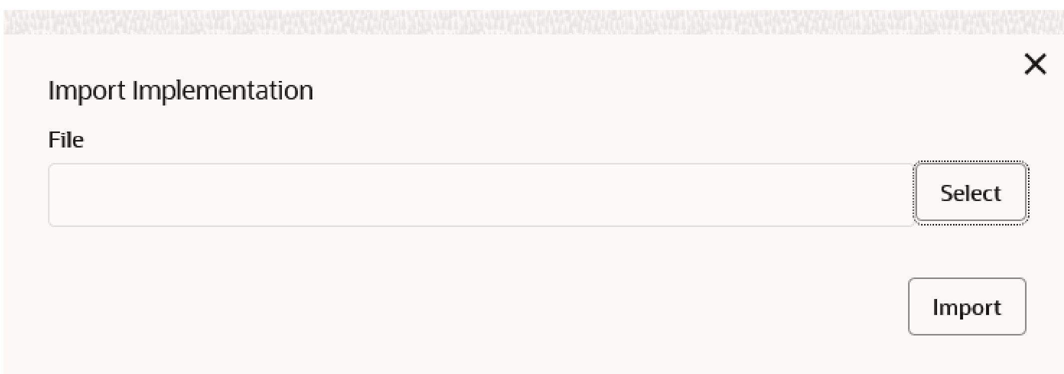
   The **Confirmation** screen displays.

Import Implementation
The user can create an implementation by importing the JSON file. The user can also import zip file in order to import all the configuration JSON files together (except parent level configuration JSON files).

6. On **Implementation** screen, click **Import**.

   The **Import Implementation** screen displays.

Figure 4-5 Import Implementation



For more information on fields, refer to the field description table.

Table 4-4 Import Implementation - Field Description

Field	Description
File	Click **Select** to select the file. **Note:** Allows only to select one file and accepts JSON and ZIP file.

7. Click **Import** to import the selected file.

   The **Confirmation** screen displays.
   The below data needs to be changed after importing provider configuration file:

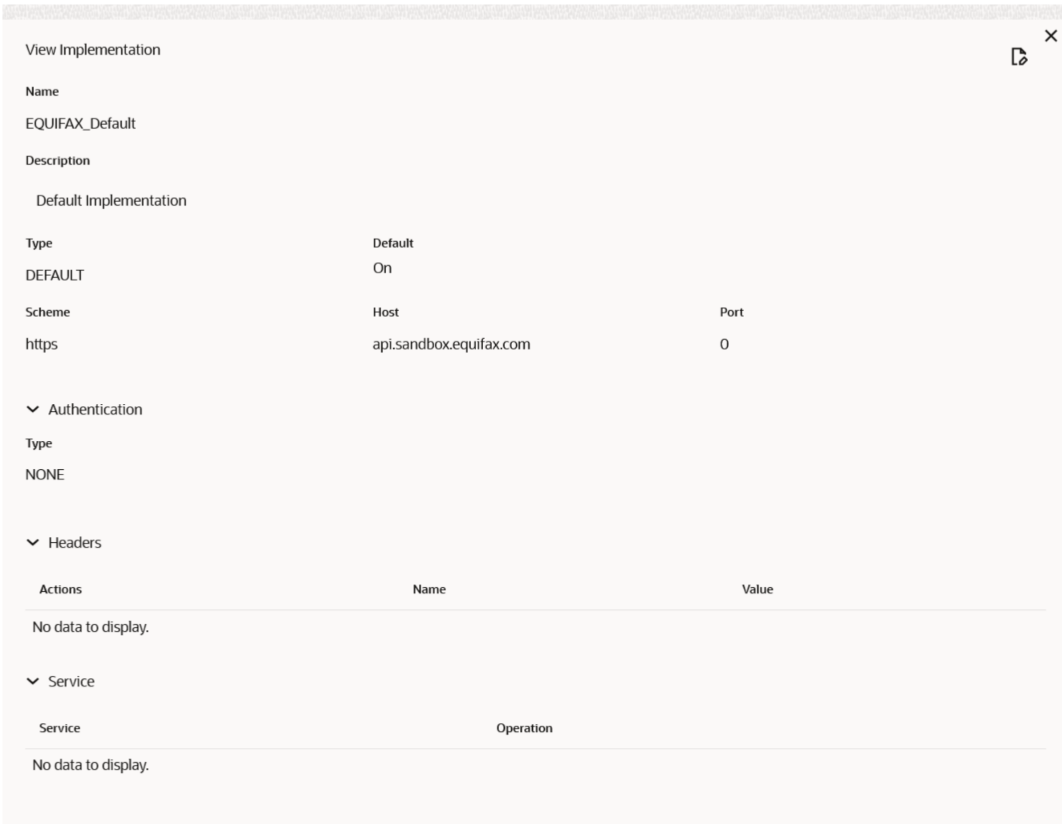
- Implementation Host and Port
- Implementation Authentication Password

View Implementation
The user can view implementation details and can also switch to edit form by clicking on edit icon.

8. On **Implementation** screen, click **Operation menu** (3 dots button) and click **View**.

   The **View Implementation** screen displays.

Figure 4-6 View Implementation

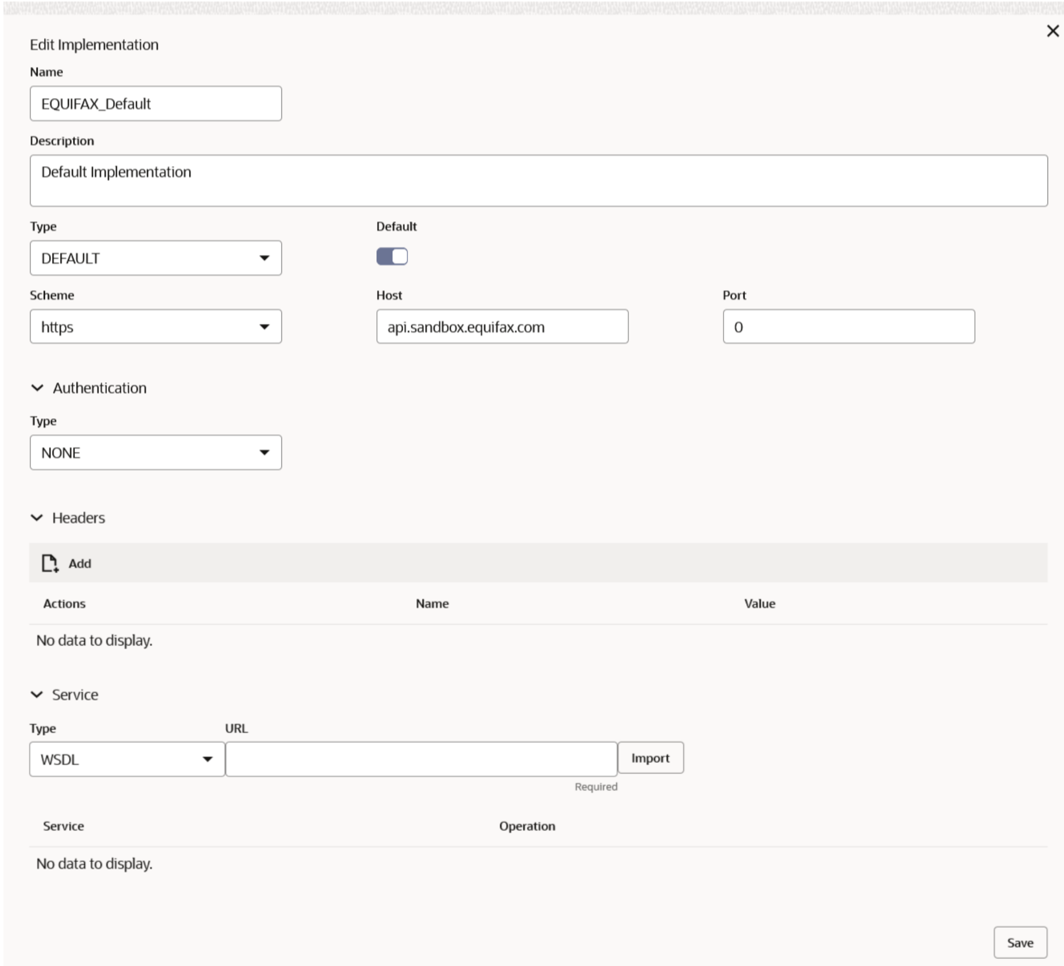


Edit Implementation
The user can modify the implementation details.

9. On **Implementation** screen, click **Operation menu** (3 dots button) and click **Edit**.

   The **Edit Implementation** screen displays.

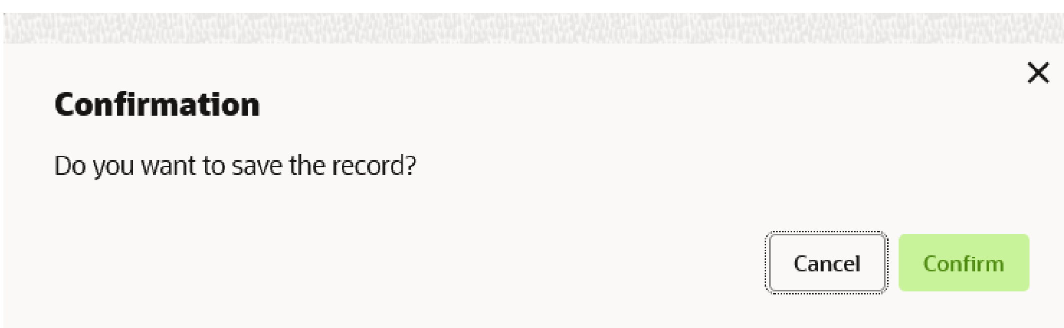
Figure 4-7 Edit Implementation



10. Click **Save** once the edit is done.

    The **Confirmation** screen displays.

Figure 4-8 Confirmation

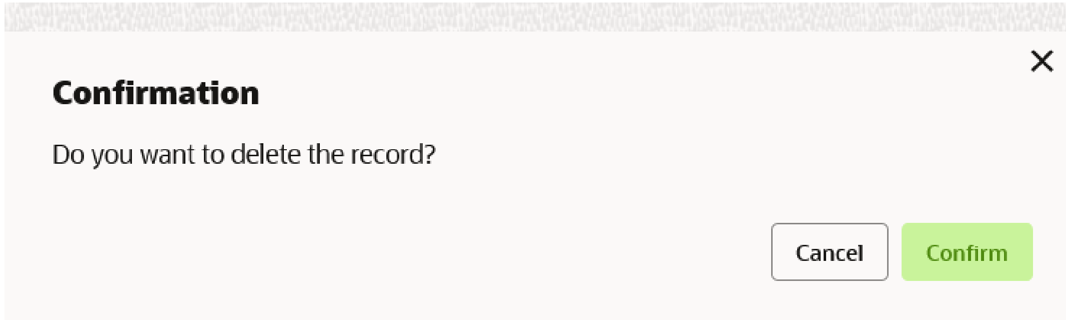


Delete Implementation
The user can delete the implementation details.

11. On **Implementation** screen, click **Operation menu** (3 dots button) and click **Delete**.

    The **Confirmation** screen displays.

Figure 4-9 Confirmation - Delete

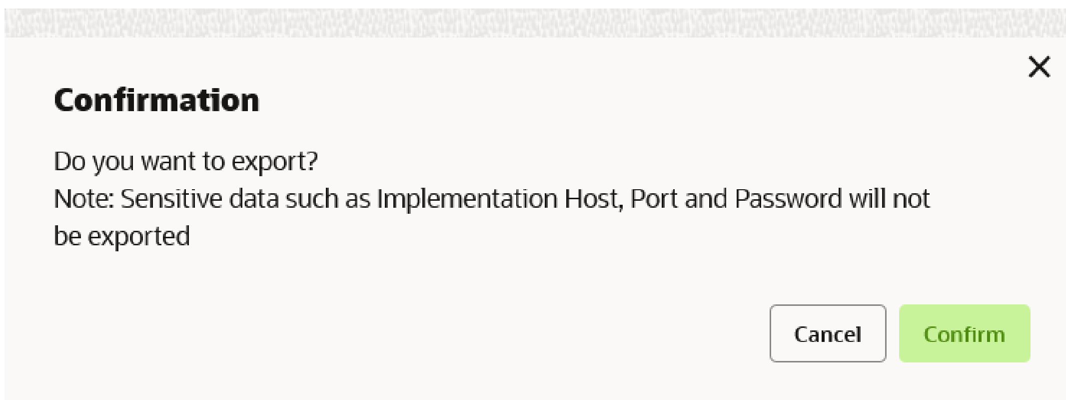


Export Implementation
The user can export the implementation configuration as JSON file.

12. On **Implementation** screen, click **Operation menu** (3 dots button) and click **Export**.

    The **Confirmation** screen displays.

Figure 4-10 Confirmation - Export Implementation



Below data cannot be exported:

- Implementation Host
- Implementation Port
- Implementation Authentication Password

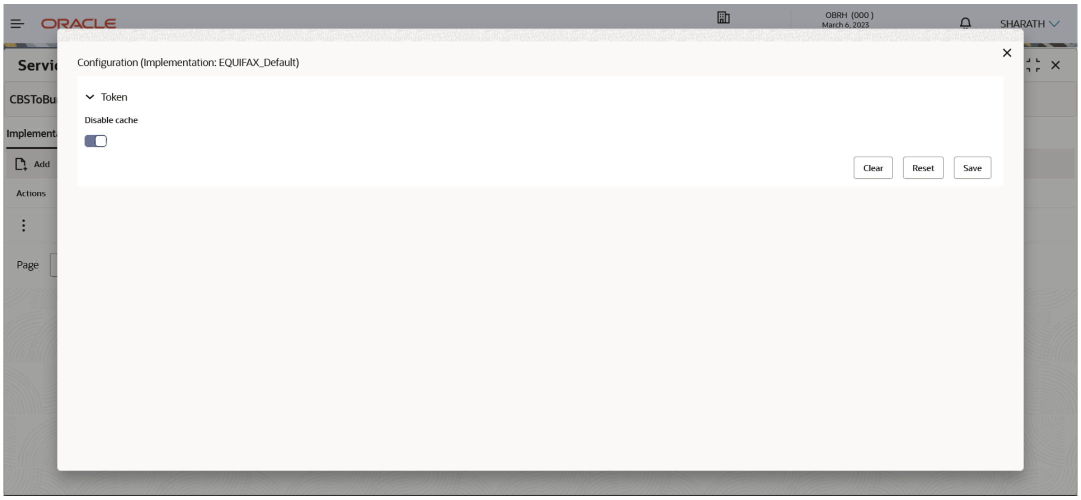
The above data needs to be configured manually after importing the configuration file. Same has been mentioned in Import section.

Configuration
End-user can configure the properties for failing the routing hub requests.

13. On **Implementation** tile, click **Operation menu** (3 dots button), and click **Configuration**.

    The **Configuration** screen displays.

Figure 4-11 Configuration



For more information on fields, refer to the field description table.

Table 4-5 Configuration - Field Description

Field	Description
Disable cache	This property is used to disable the token caching. **Note:** Default value is false.

Request Audit

14. On **Implementation** screen, click **Operation menu** (3 dots button) and click **Request Audit**.

    The **Request Audit** screen displays.

    > **Note:**
    >
    > Refer to Request Audit topic for screen and field description.

Clear Cache
The user can clear the SOAP client cache.

15. On **Implementation** screen, click **Operation menu** (3 dots button) and click **Clear Cache**.

5 Consumer Services

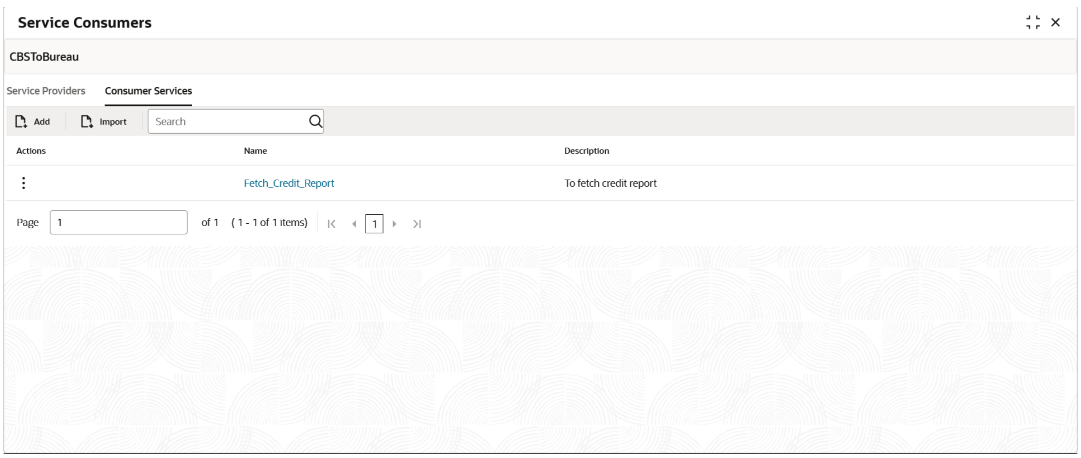
This topic describes the systematic instructions to configure the consumer services.

The **Consumer Services** defines the service ID, which sends from the service consumer. It also caters the transition and route definition. It comprises of source integration details.

1. On **Service Consumers** screen, click **Consumer Services**.

   The **Consumer Services** screen displays.

Figure 5-1 Consumer Services

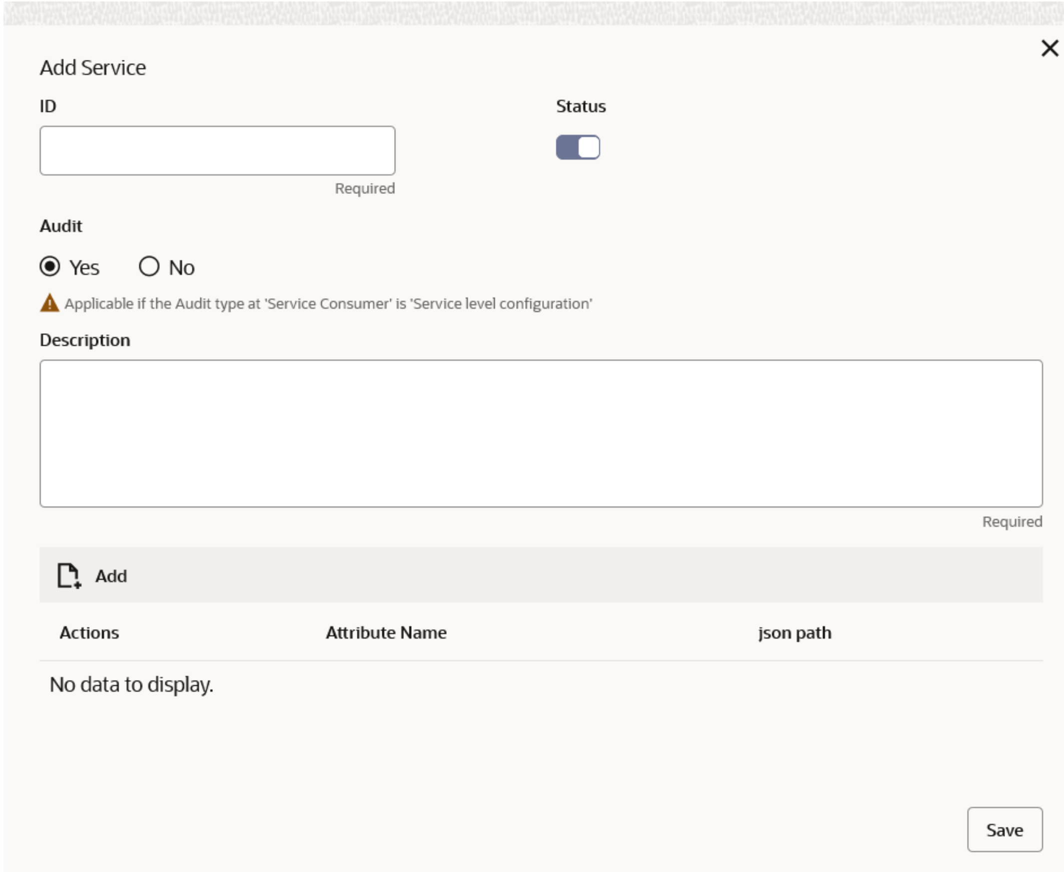


Add Consumer Service
The user can create Consumer Service manually.

2. On **Consumer Services** screen, click **Add**.

   The **Add Service** screen displays.

Figure 5-2 Add Service



3. Specify the fields on **Add Service** screen.

> **Note:**
>
> The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

Table 5-1 Add Service - Field Description

Field	Description
ID	Specify the ID of the consumer service. **Note:** • Enter 0 to maximum of 255 characters. • No numeric value at beginning and no space allowed.

Table 5-1 (Cont.) Add Service - Field Description

Field	Description
Status	Active / Inactive If consumer service is marked as inactive, then all related routes will be stopped.
Audit	Select the Audit option for the consumer service. The available options are: • **Yes** - This option is for enabling the audit for consumer service. • **No** - This option is for disabling the audit for consumer service. **Note:** This option is only applicable if audit type at 'Service Consumer' is 'Service level configuration'
Description	Specify the description of the consumer service. **Note:** • Enter 0 to 1000 characters. • No space allowed at beginning or ending of the characters.
Add	To add, refer to the below step.
Actions	Displays the action. The user can edit or delete the header.
Attribute Name	Displays the name of the attribute.
json path	Displays the JSON path.

4. To add **Attributes**, follow the below steps.
   a. Click **Add**.

      The **Add Attribute** screen displays.

Figure 5-3   Add Attribute

b. Specify the fields on **Add Attribute** screen.

> **Note:**
>
> The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

Table 5-2   Add Header - Field Description

Field	Description
Name	Specify the name of the attribute.
josn path	Specify the json path.

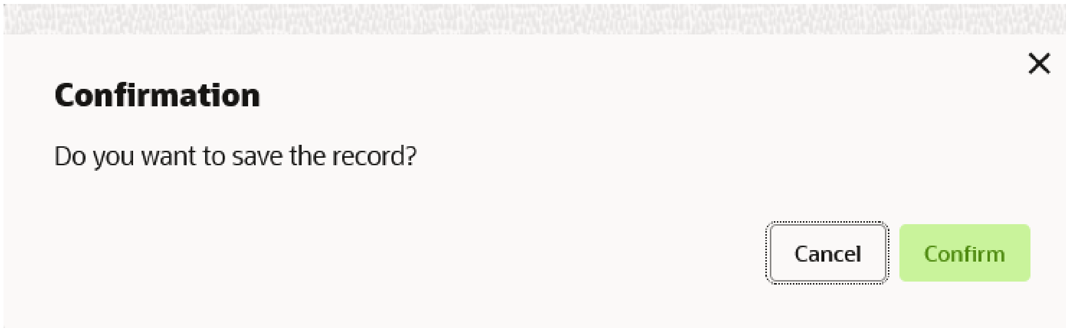
> **Note:**
>
> - Using $.body, the user can access the request body.
>   Syntax: $.body.fieldName
>
>   Example: $.body.branchCode
> - Using $.headers, the user can access the request headers.
>   Syntax: $.headers["fieldname "][0]
>
>   Example: $.headers["branchCode"][0]
> - Using $.env, the user can access the environment variables.
>   Syntax: $.env.group.variable

c. Click **OK** to save the attributes.

5. Click **Save** to save the details.

   The **Confirmation** screen displays.

   Figure 5-4 Confirmation

6. On **Confirmation** screen, click **Confirm** to add the service.

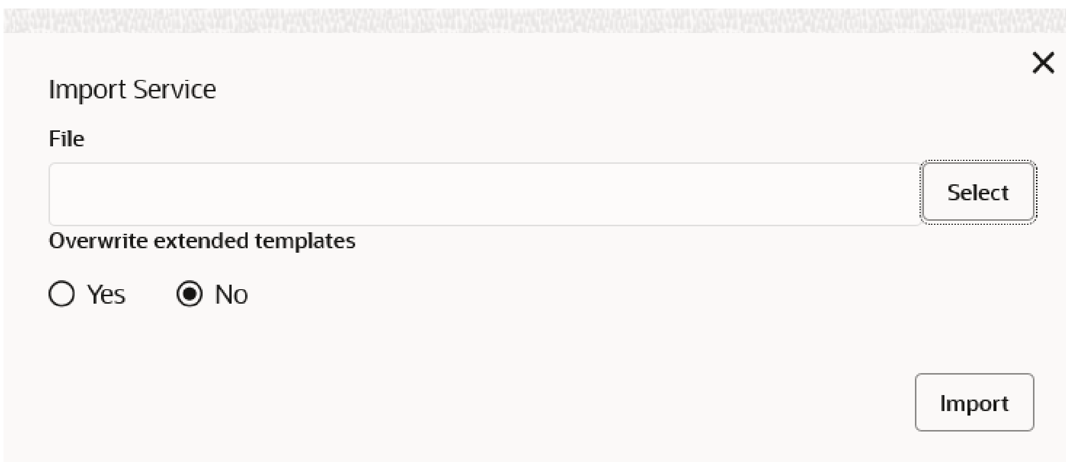
Import Consumer Service
The user can create a consumer service by importing the JSON file.

The user can also import zip file in order to import all the configuration JSON files together (except parent level configuration JSON files).

7. On **Consumer Services** screen, click **Import**.

   The **Import Service** screen displays.

   Figure 5-5 Import Service

   For more information on fields, refer to the field description table.

Table 5-3 Import Service - Field Description

Field	Description
File	Select the file using **Select** button. **Note:** Allows only to select one file and accepts only JSON file.
Overwrite extended templates	Select the respective radio button to overwrite the extended templates. The available options are: • Yes - This option overwrites the extended templates. • No - This option retains the existing extended templates.

8. Click **Import** to import the selected file.

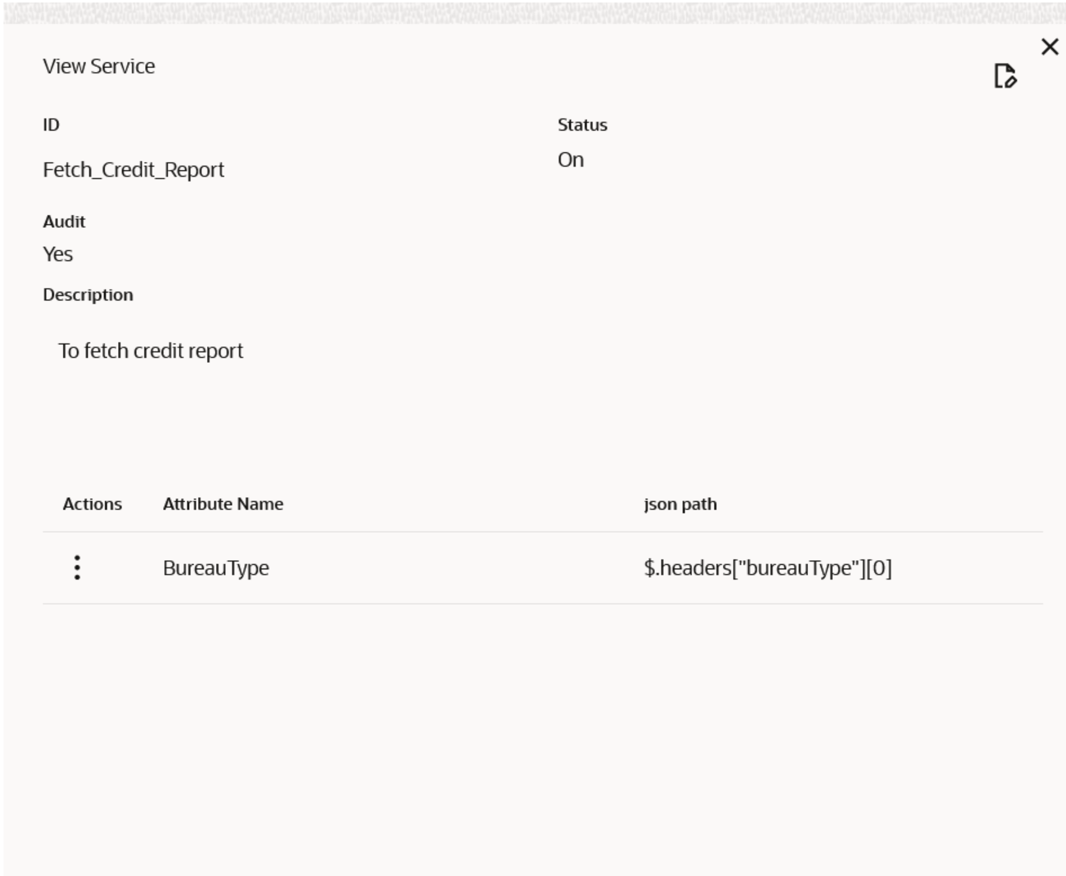
   The **Confirmation** screen displays.

View Consumer Service
The user can view consumer service details and can also switch to edit form by clicking on edit icon.

9. On **Consumer Service** tile, click **Operation menu** (3 dots button), and click **View**.

   The **View Service** screen displays.

Figure 5-6 View Service

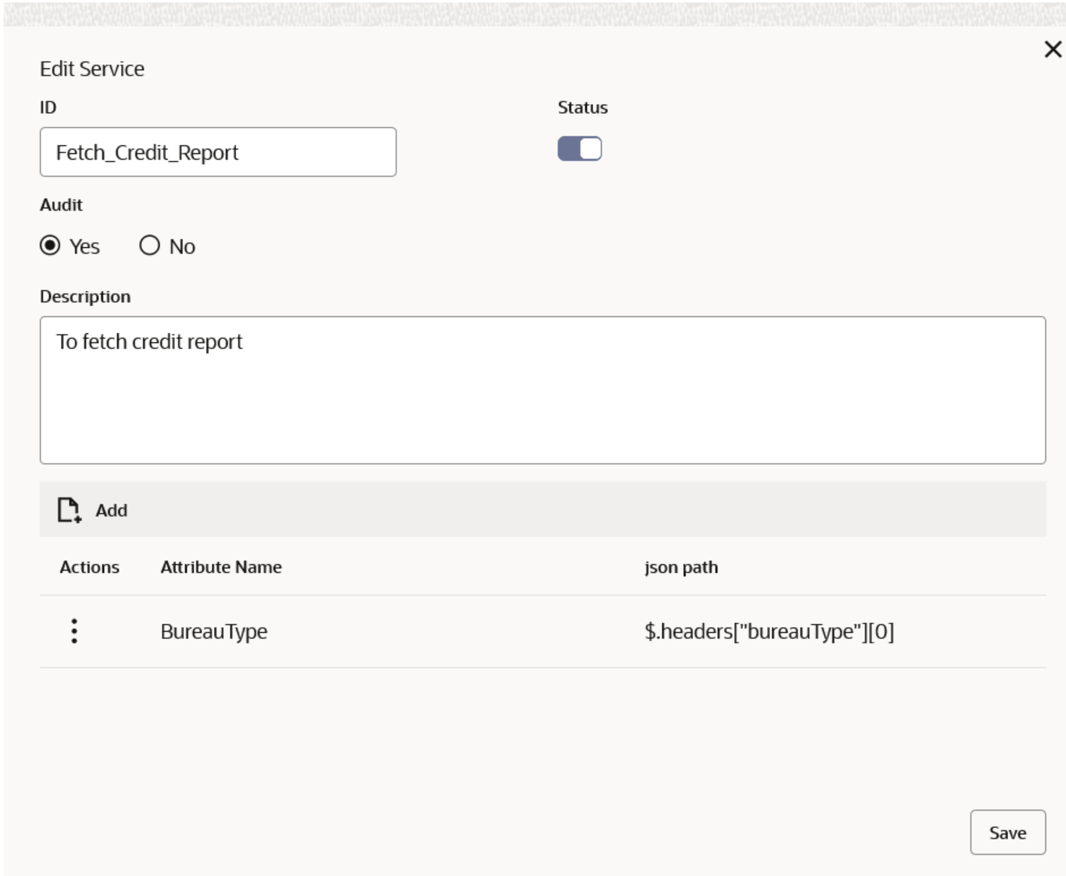


Edit Consumer Service
The user can modify the consumer service details.

10. On **Consumer Service** tile, click **Operation menu** (3 dots button), and click **Edit**.

    The **Edit Service** screen displays.

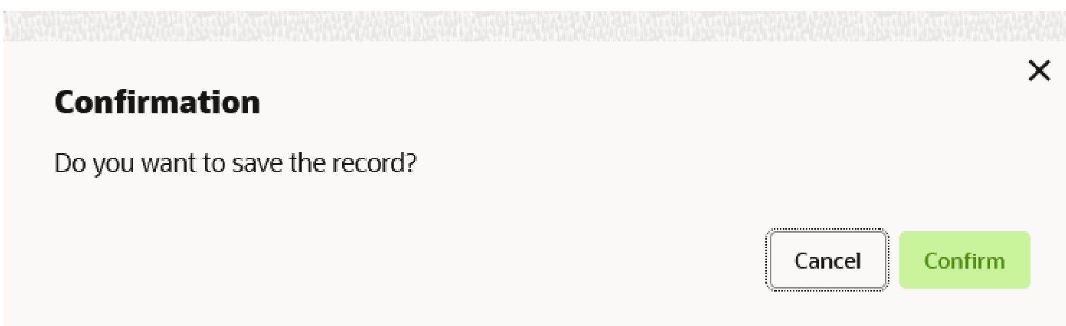
Figure 5-7 Edit Service



11. Click **Save** once the edit is done.

    The **Confirmation** screen displays.

Figure 5-8 Confirmation - Edit



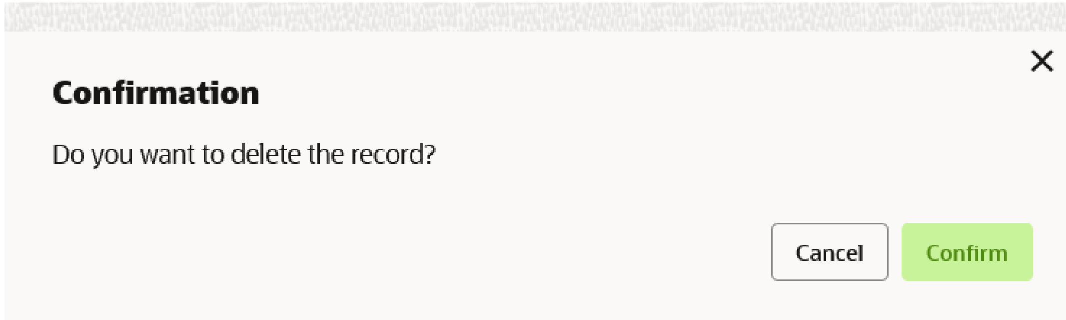
Delete Consumer Service

The user can delete the consumer service.

12. On **Consumer Service** tile, click **Operation menu** (3 dots button), and click **Delete**.

    The **Confirmation** screen displays.

Figure 5-9 Confirmation

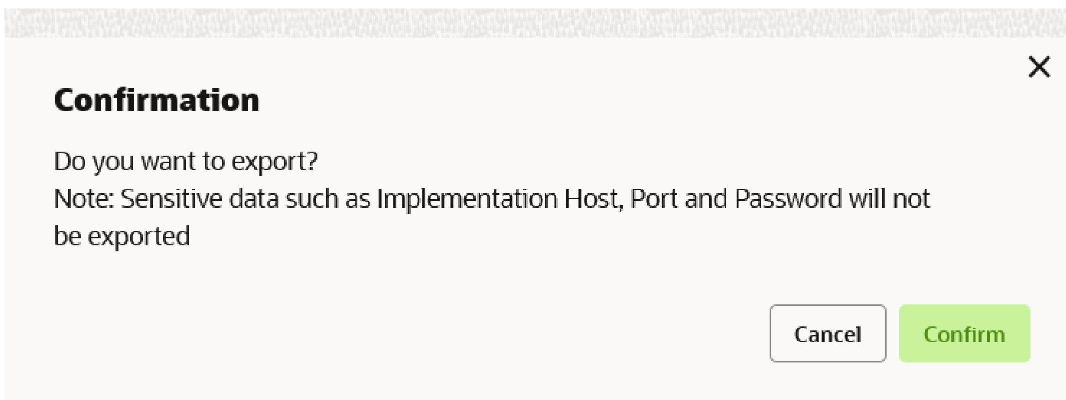


Export Consumer Service
The user can export the consumer service configuration as JSON file.

13. On **Consumer Service** tile, click **Operation menu** (3 dots button), and click **Export**.

    The **Confirmation** screen displays.

Figure 5-10 Confirmation - Export



Consumer Service - Configuration

14. On **Consumer Service** tile, click **Operation menu** (3 dots button), and click **Configuration**.

    The **Configuration** screen displays.

    > **Note:**
    >
    > Refer to Configuration topic for the screen and field description.

Consumer Service - Request Audit

15. On **Consumer Service** tile, click **Operation menu** (3 dots button), and click **Request Audit**.

    The **Request Audit** screen displays.

> **Note:**
>
> Refer to Request Audit topic for the screen and field description.

6
Transformation

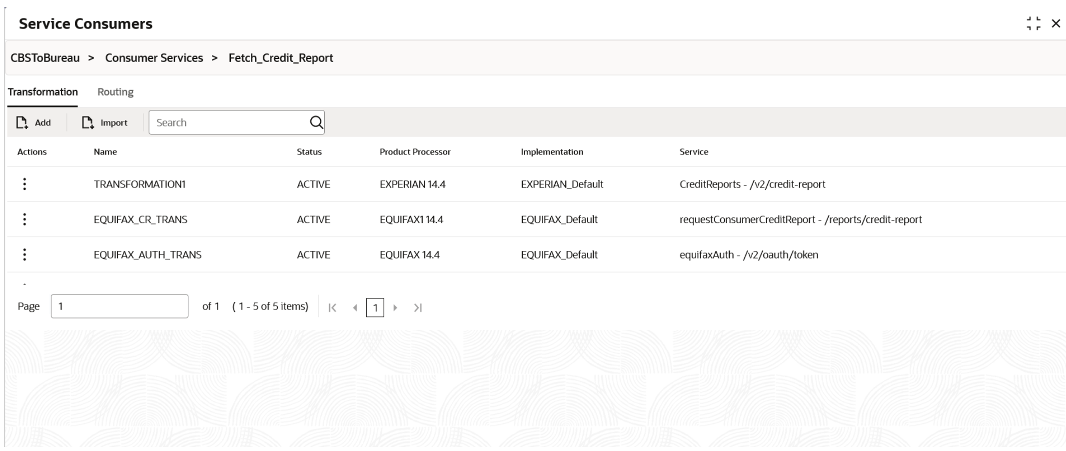
This topic describes the systematic instructions to configure the transformation.

The **Transformation** acts as assembling and transforming data from source to destination and vice-versa. This takes place under consumer service. This converts the data of service consumer into service provider.

1. On **Consumer Services** screen, click the required consumer service tile.

   The **Transformation** screen displays.

   Figure 6-1 Transformation

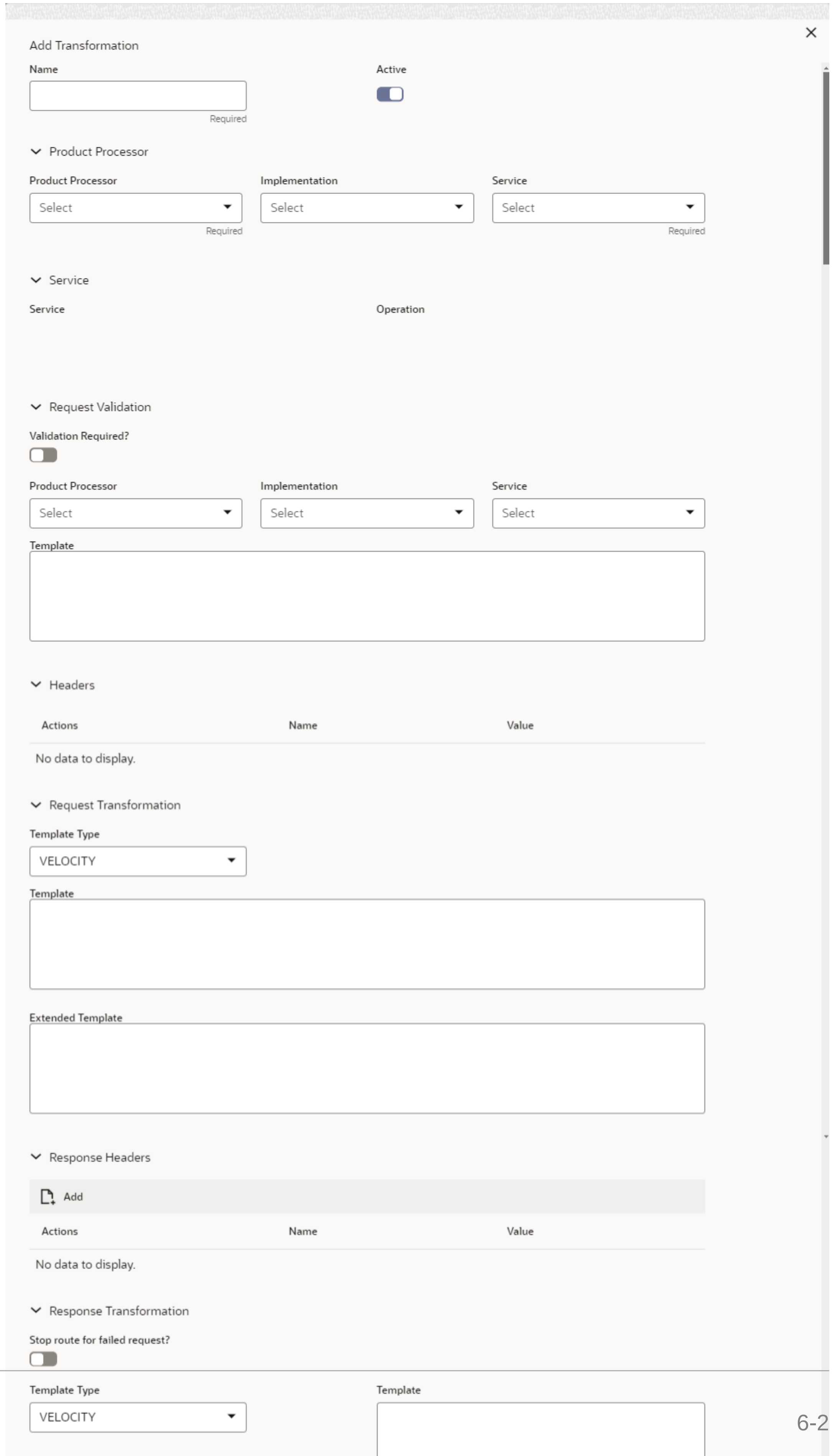
   

Add Transformation
The user can create transformation manually.

2. On **Transformation** screen, click **Add**.

   The **Add Transformation** screen displays.

Figure 6-2 Add Transformation



3. Specify the fields on **Add Transformation** screen.

> **Note:**
>
> The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

Table 6-1 Add Transformation - Field Description

Field	Description
Name	Specify the name for the transformation. **Note:** • Enter 0 to maximum of 255 characters. • No numeric value at beginning and no space allowed.
Active	Active / Inactive If transformation is marked as inactive, then the user will not be able to select transformation in routing.
Product Processor	Displays the Product Processor details.
Product Processor	Select the product processor from the drop-down list.
Implementation	Select the implementation from the drop-down list.
Service	Select the service from the drop-down list.
Service	Displays the service details of the selected service.
Operation	Displays the operation details of the selected service.
Request Validation	Displays the Request Validation details.
Validation Required?	Select the toggle to enable the validation required for request. **Note:** Validation Model of Oracle Banking Pricing & Decision Service is only supported.
Product Processor	Select the product processor from the drop-down list.
Implementation	Select the implementation from the drop-down list.
Service	Select the service from the drop-down list.
Template	Specify the template in which validation provider accepts.
Headers	Displays the header list relevant to the selected provider, implementation and service. User can change the header values. The value can either be hardcoded or can be Velocity mapping.
Path Params	Displays the path param list relevant to the selected service. User can change the param values. Value can either be hardcoded or can be Velocity mapping.

Table 6-1 (Cont.) Add Transformation - Field Description

Field	Description
Query Params	Displays the query param list relevant to the selected service. User can change the param values. Value can either be hardcoded or can be Velocity mapping.
Request Transformation	Displays the Request Transformation details.
Body Type	Select the body type for the Request Transformation from the drop-down list. The available options are: • **RAW** • **FORM DATA** • **BINARY** **Note:** This field appears only if the selected service is REST service and **RAW** option is used for URL-encoded content type.
Template Type	Select the template type for the Request Transformation from the drop-down list. The available options are: • **VELOCITY** • **JSLT** • **XSLT**
Template	Specify the template for the Request Transformation in which provider accepts. Refer to Transformation Type for syntax.
Extended Template	Specify the custom template in order to extend the kernel template. Refer to Extensibility and Transformation Type for syntax. **Note:** This field appears only if the **Body Type** is selected as **RAW**.
Response Header	Specify the additional headers required to be part of Routing Hub response headers. Value can either be hardcoded or can be Velocity mapping.
Response Transformation	Displays the response transformaton details.
Stop route for failed request	This property is used to handle response for failed request. **Note:** Only applicable for API chaining scenario.

Table 6-1 (Cont.) Add Transformation - Field Description

Field	Description
Template Type	Select the template type for the Response Transformation from drop-down list. The available options are: • **VELOCITY** • **JSLT** • **XSLT**
Template	Specify the kernel template in which consumer accepts. Refer to Transformation Type for syntax.
Mocking required?	Select the toggle if the mocking is required for the Response Transformation or not. If the toggle is **ON**, the Routing Hub will return the mocked template output (with extended template output if mentioned) to consumer without invoking provider API.
Mock Template	Specify the kernel template for the Response Transformation in which the consumer accepts. Refer Transformation Type for syntax.

4. Click **Save** to save the details.

   The **Confirmation** screen displays.

5. Click **Confirm** to add the transformation.

Import Transformation
The user can create a transformation by importing the JSON file. The user can also import zip file in order to import all the configuration JSON files together (except parent level configuration JSON files).

6. On **Transformation** screen, click **Import**.

   The **Import Transformation** screen displays.

Figure 6-3 Import Transformation

For more information on fields, refer to the field description table.

Table 6-2 Import Transformation - Field Description

Field	Description
File	Select the file using **Select** button. **Note:** Allows only to select one file and accepts JSON and ZIP file.
Overwrite extended templates	Select the respective radio button to overwrite the extended templates. The available options are: • Yes - This option overwrites the extended templates. • No - This option retains the existing extended templates.

7. Click **Import** to import the selected file.

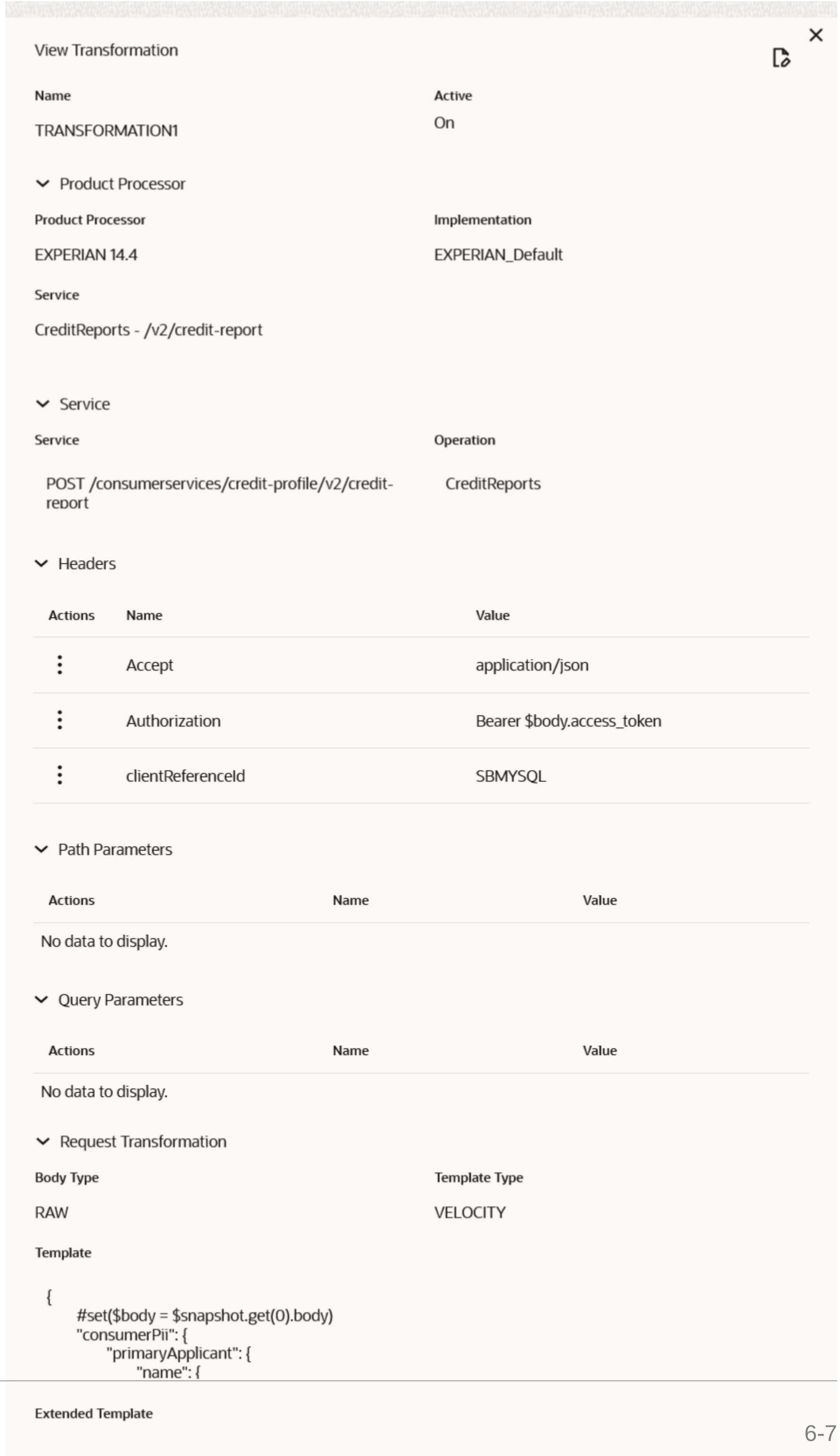
   The **Confirmation** screen displays.

View Transformation
The user can view transformation details and can also switch to edit form by clicking on edit icon.

8. On **Transformation** list, click **Operation menu** (3 dots button), and click **View**.

   The **View Transformation** screen displays.

Figure 6-4 View Transformation



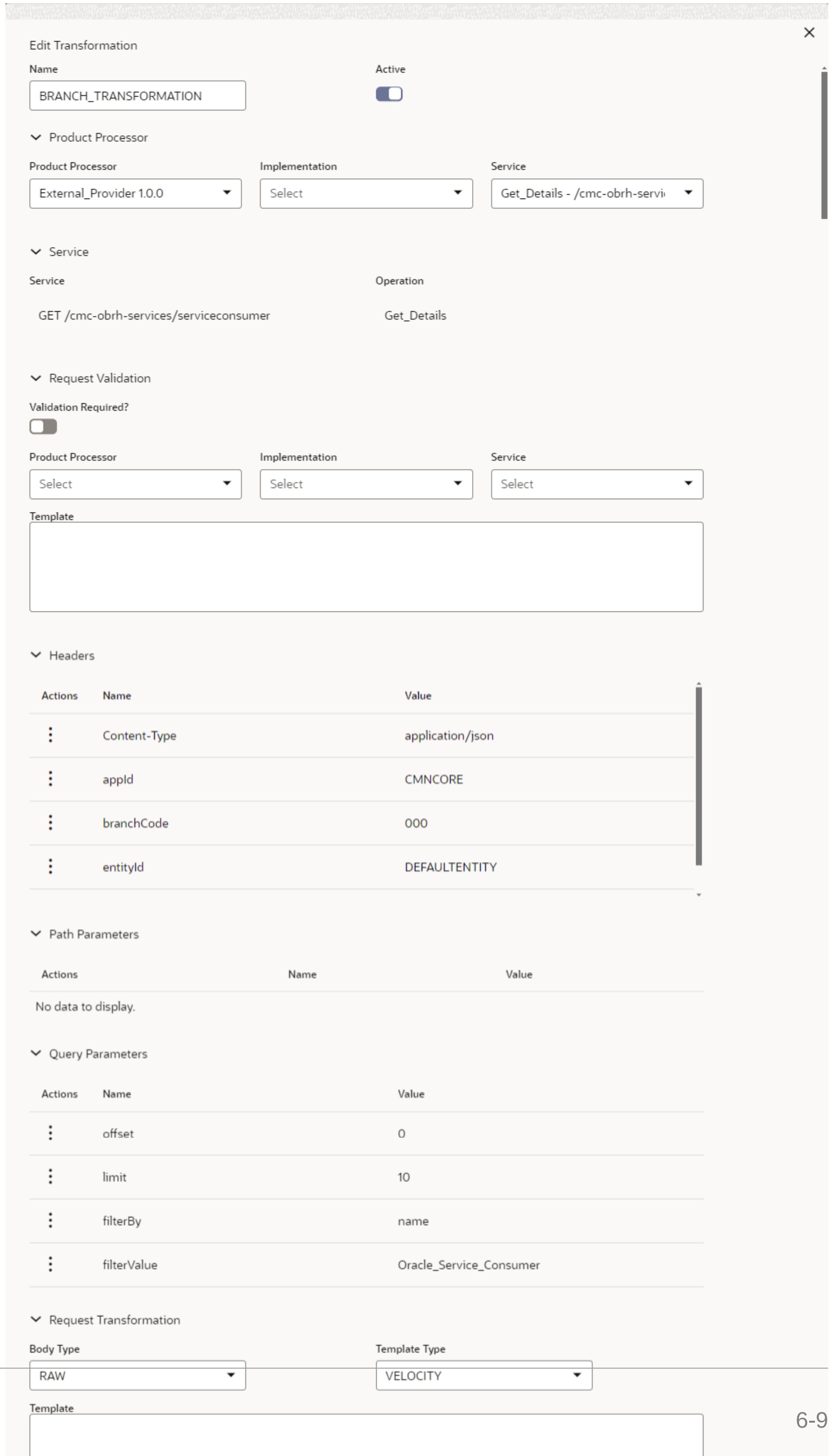
Click **Edit** icon to edit the **Transformation**.

Edit Transformation
The user can modify the transformation details.

9. On **Transformation** list, click **Operation menu** (3 dots button), and click **Edit**.

   The **Edit Transformation** screen displays.

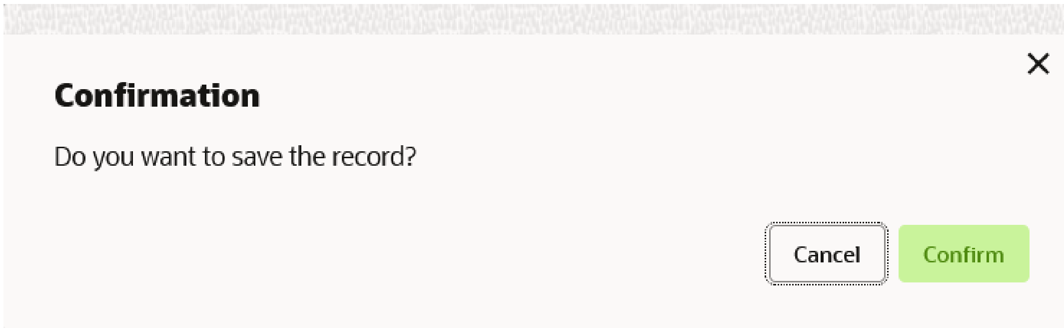
Figure 6-5 Edit Transformation



10. Click **Save** once the edit is done.

    The **Confirmation** screen displays.

    Figure 6-6 Confirmation

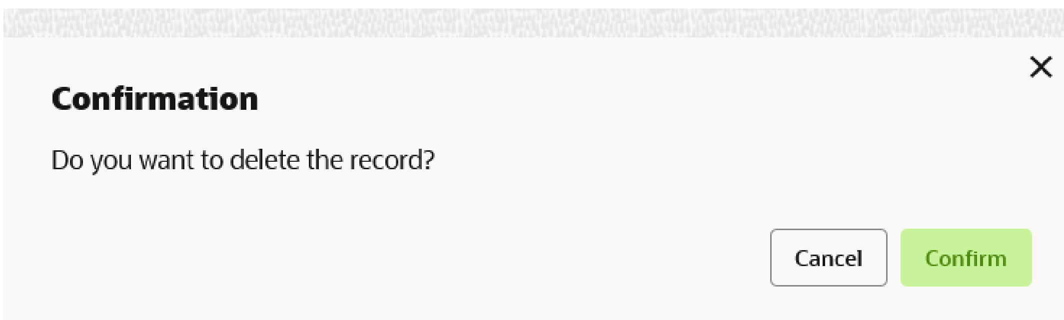
    

Delete Transformation
The user can delete the transformation.

11. On **Transformation** list, click **Operation menu** (3 dots button), and click **Delete**.

    The **Confirmation - Delete** screen displays.

    Figure 6-7 Confirmation - Delete

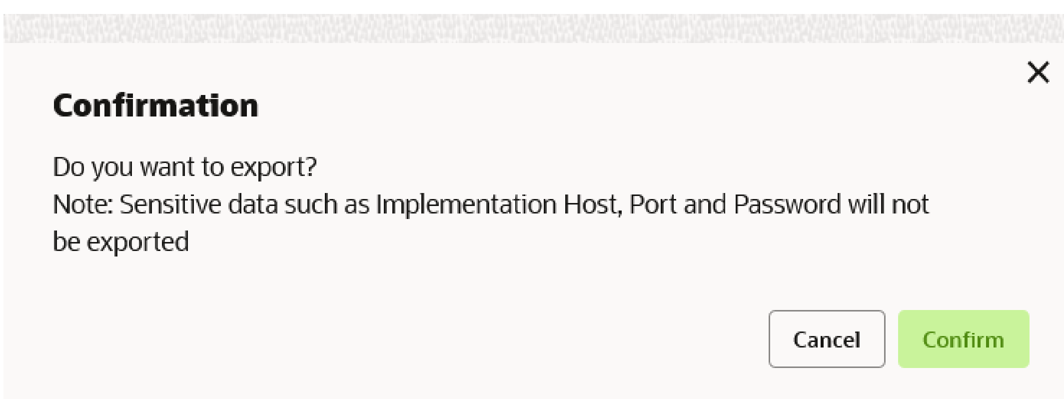
    

Export Transformation
The user can export the transformation configuration as JSON file.

12. On **Transformation** list, click **Operation menu** (3 dots button), and click **Export**.

    The **Confirmation** screen displays.

    Figure 6-8 Confirmation - Export

Request Audit

13. On **Transformation** list, click **Operation menu** (3 dots button), and click **Request Audit**.

    The **Request Audit** screen displays.

    > **Note:**
    >
    > Refer to Request Audit topic for screen and field description.

7
Routing

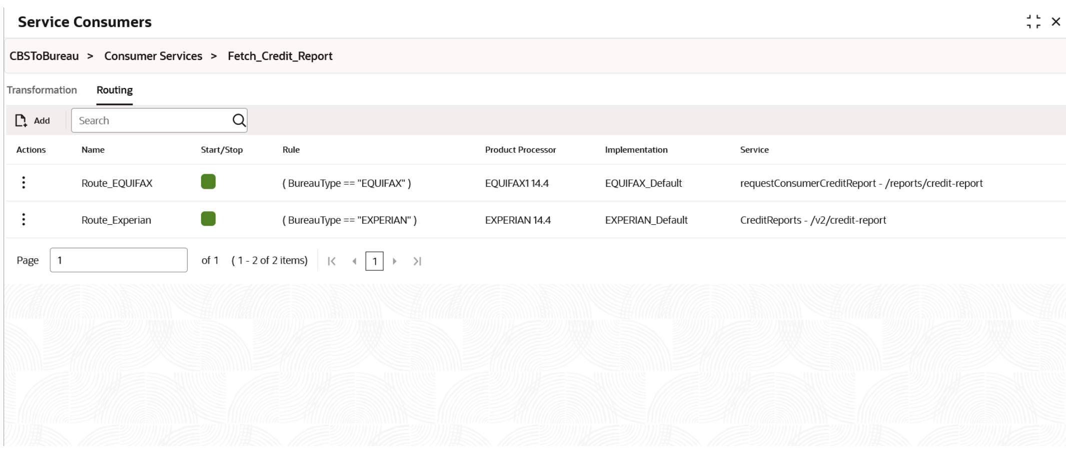
This topic describes the systematic instructions to configure the routing.

Routing defines no rule or rule-based route configuration. Route decide the actual request to be send to which service provider based on maintenance and evaluation.

1. On **Consumer Services** screen, click **Routing**.

   The **Routing** screen displays.

Figure 7-1 Routing



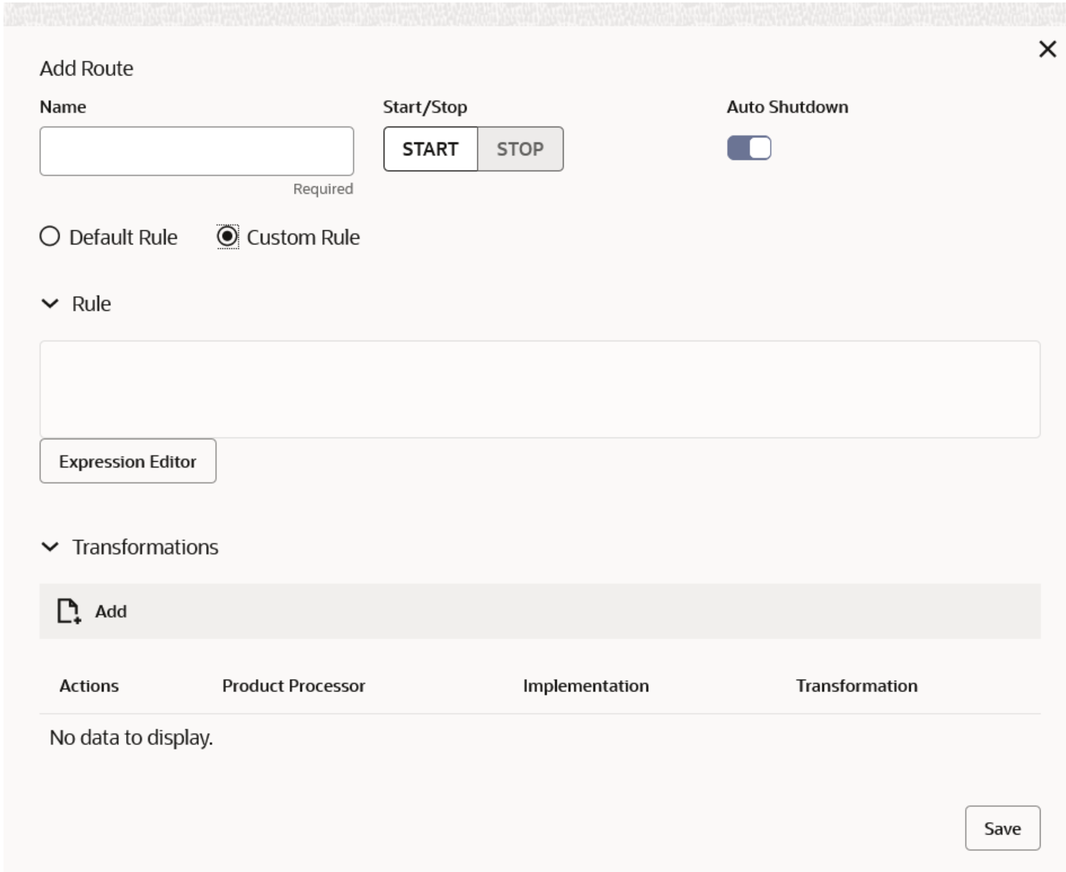
Add Route

The user can create routing manually.

2. On **Routing** screen, click **Add**.

   The **Add Route** screen displays.

Figure 7-2 Add Route



3. Specify the fields on **Add Route** screen.

> **Note:**
>
> The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

Table 7-1 Add Route - Field Description

Field	Description
Name	Specify the name for the route. **Note:** • Enter 0 to maximum of 255 characters. • No numeric value at beginning and no space allowed.
START / STOP	START / STOP If routing is marked as STOP, then consumer request fails at routing hub level only.

Table 7-1 (Cont.) Add Route - Field Description

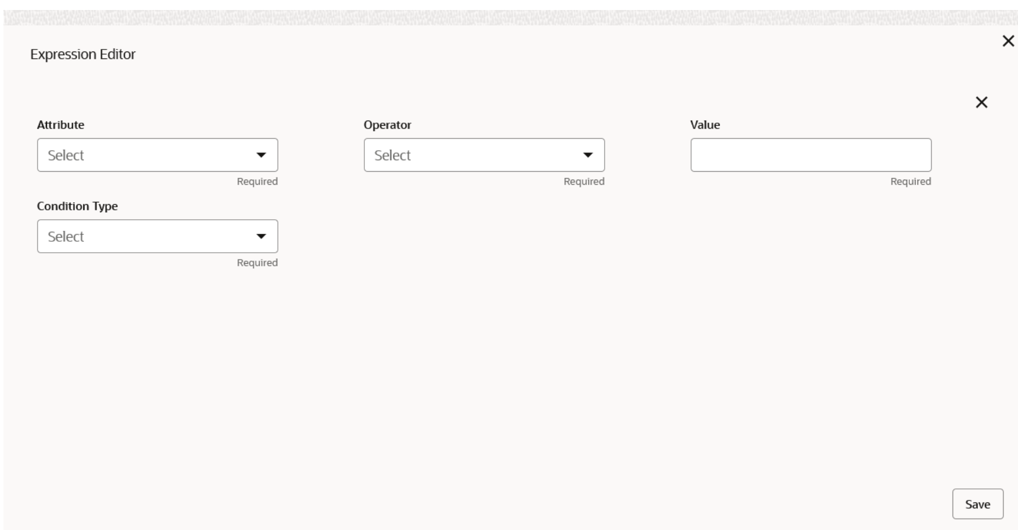
Field	Description
Auto Shutdown	ON / OFF If AutoShutdown flag is ON, then route state will be changed to STOP if route failure goes beyond the threshold failure limit based on the monitoring and alert configuration.
Rule Type	Select the rule type. The available options are: • **Default Rule** • **Custom Rule**
Expression Editor	Displays the expression that is formed through expression editor.
Add	To add, refer to the below steps.
Actions	Displays the action. The user can edit or delete the header.
Product Processor	Displays the product processor.
Implementation	Displays the implementation.
Transformation	Displays the transformation.

Add Custom Rule using Expression Editor

4. To add **Editor**, follow the below steps.
   a. On **Add Route** screen, click **Editor** button.

      The **Expression Editor** screen displays.

      Figure 7-3 Expression Editor

   b. Specify the fields on **Expression Editor** screen.

      > **Note:**
      >
      > The fields marked as **Required** are mandatory.

      For more information on fields, refer to the field description table.

Table 7-2 Expression Editor - Field Description

Field	Description
Attribute	Select attribute relevant to consumer service from drop-down list.
Operator	Select the logical operators to form an expression from drop-down list.
Value	Specify the value. **Note:** Enter 0 to 255 characters.
Condition Type	Select the condition type from drop-down list.

c. Click **Save** to save the details.

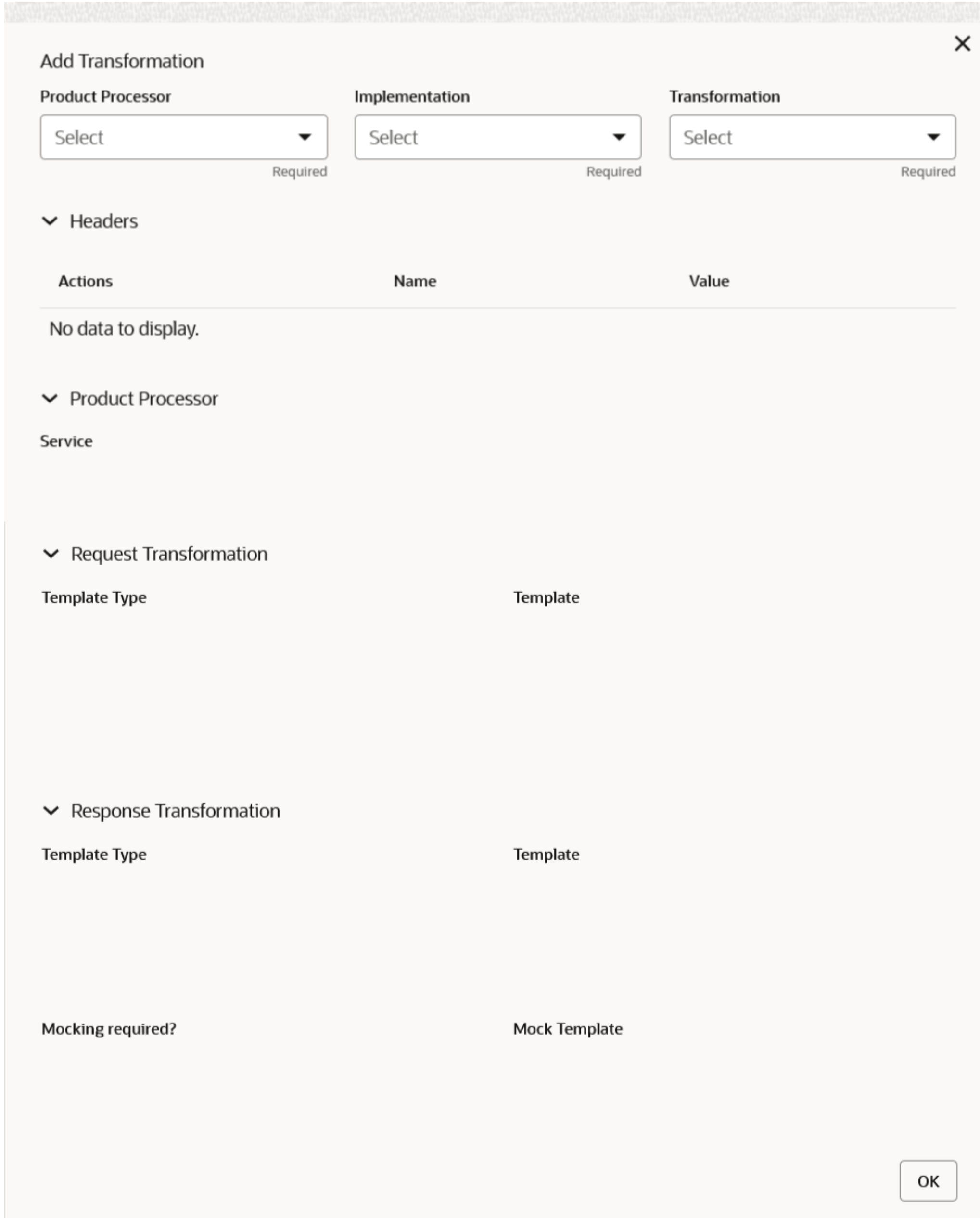
> **Note:**
>
> String value should be quoted using single quotes (') Example: 'abc'
> List value should be comma separated values and quoted using single quotes (') Example: 'abc,xyz,1.23,true'

Transformations
The user can define the sequence of transformations for each routing in which request should be processed. Sequence of transformations in list can be changed by using drag-n-drop feature.

5. To add **Transformations**, follow the below steps.
   a. On **Add Route** screen, click **Add**.

      The **Add Transformation** screen displays.

Figure 7-4 Add Transformation



b. Specify the fields on **Add Transformation** screen.

> **Note:**
>
> The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

Table 7-3 Add Transformation - Field Description

Field	Description
Product Processor	Select the product processor from the drop-down list.
Implementation	Select the implementation from the drop-down list.
Transformation	Select the transformation from the drop-down list.
Action	Displays the action. The user can edit or delete the header.
Name	Displays the name of the header.
Value	Displays the value of the header.
Service	Displays the service of the product processor.
Template Type	Displays the template type for the request transformation.
Template	Displays the template for the request transformation.
Template Type	Displays the template type for the response transformation.
Template	Displays the template for the response transformation.
Mocking required?	Displays whether the mocking required for the response transformation or not.
Mock Template	Displays the mock template for the response transformation.

   c. Click **OK**.

6. Click **Save** to save the details.

   The **Confirmation** screen displays.

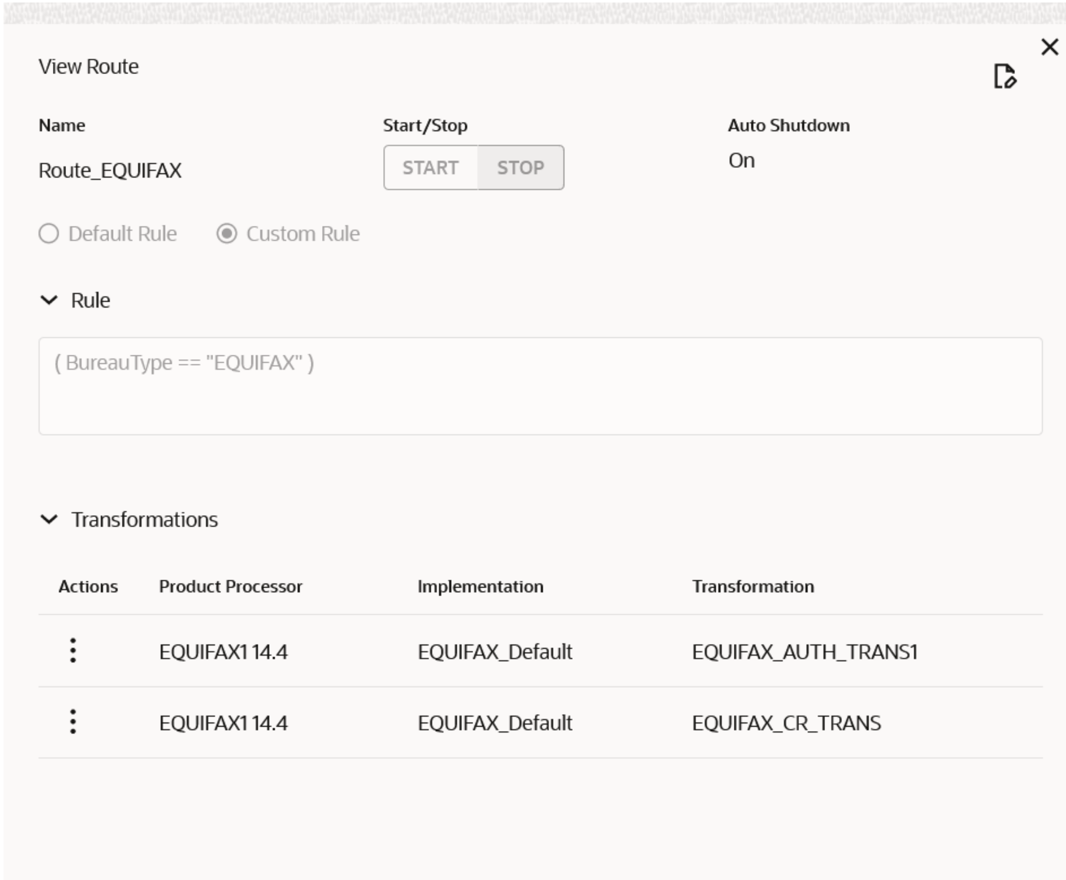
7. Click **Confirm** to add the routing details.

View Route
The user can view the routing details and can also switch to edit form by clicking on edit icon.

8. On **Routing** screen, click **Operation menu** (3 dots button), and click **View**.

   The **View Route** screen displays.

Figure 7-5 View Route

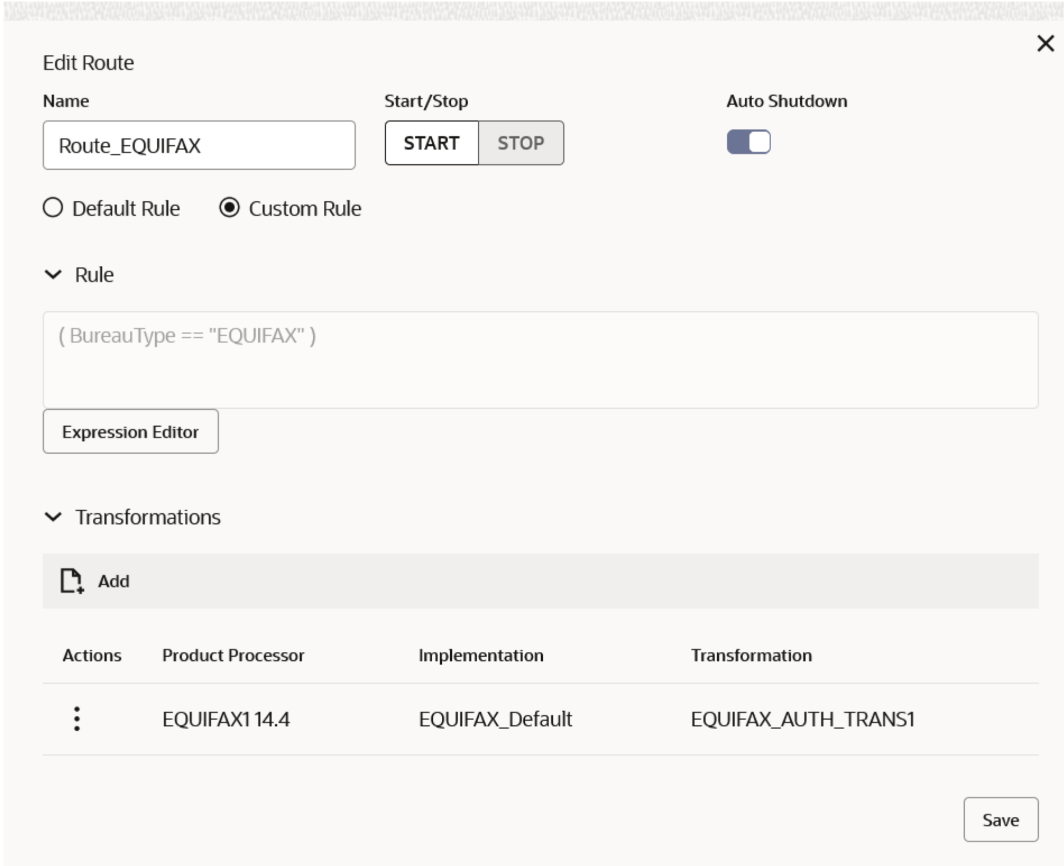


Edit Route
The user can modify the routing details.

9. On **Routing** screen, click **Operation menu** (3 dots button), and click **Edit**.

   The **Edit Route** screen displays.

Figure 7-6   Edit Route



Delete Route

The user can delete the routing details.

10. On **Routing** screen, click **Operation menu** (3 dots button), and click **Delete**.

    The **Confirmation** screen displays.

Figure 7-7   Confirmation - Delete

11. Click **Confirm** to delete the selected routing.

Routing - Configuration

12. On **Routing** screen, click **Operation menu** (3 dots button), and click **Configuration**.

    The **Configuration** screen displays.

> **Note:**
>
> Refer to Configuration topic for screen and field description.

Routing - Request Audit

13. On **Routing** screen, click **Operation menu** (3 dots button), and click **Request Audit**.

    The **Request Audit** screen displays.

> **Note:**
>
> Refer to Request Audit topic for screen and field description.

8 Chaining

This topic provides the information about chaining of the transformation.

The end-user can define the sequence of transformations for each routing in which the request needs to be processed.

Chaining can be achieved by using the snapshot list. The snapshot list stores the response body and response headers whenever the transformation is processed. Therefore, the end-user can access the response body or headers of all processed transformations at any stage.

Syntax: $snapshot.get(index).body or $snapshot.get(index).headers

> **Note:**
>
> $body and $headers refers to the response body and headers of previous step.

Figure 8-1 Chaining

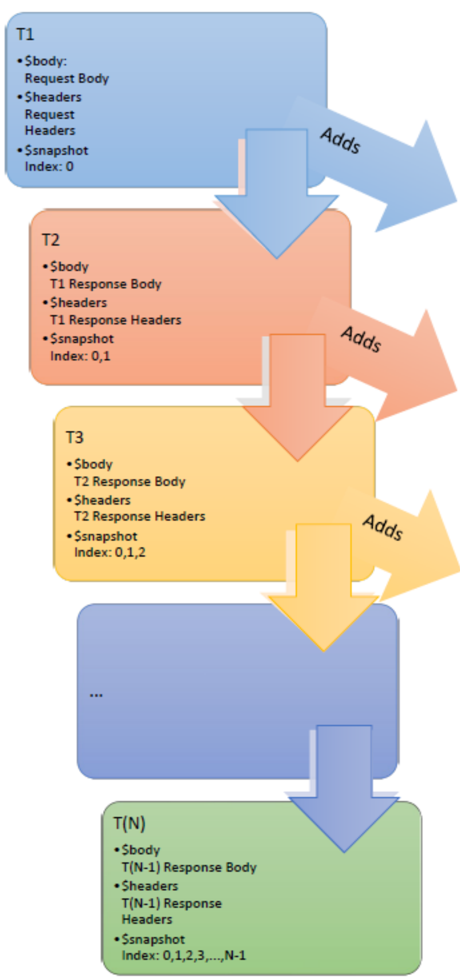


Table 8-1 Snapshot List

Index	Body	Headers
1	Request Body	Request Headers
2	T1 Response Body	T1 Response Headers
3	T2 Response Body	T2 Response Headers
4	T3 Response Body	T3 Response Headers
…	...	...
N	T(N-1) Response Body	T(N-1) Response Headers

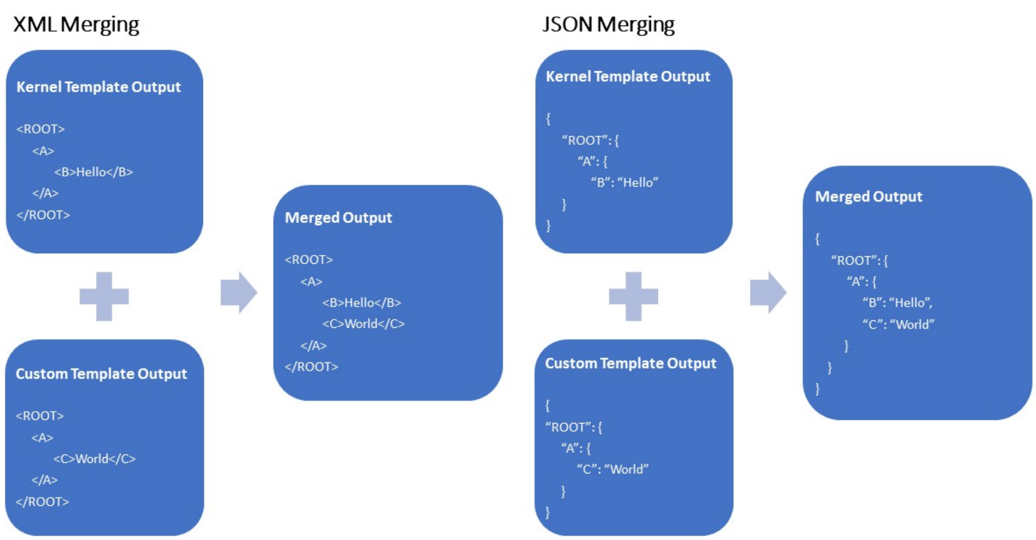
9 Template Extensibility

Template Extensibility in Routing Hub refers to template extensibility and is achieved by specifying the extended templates for request and response kernel transformation templates. And as part of extensibility, Routing Hub merges the output of kernel template and custom template in terms of JSON / XML merging.

In case of request, Routing Hub will send the merged output as request payload to provider.

In case of response, Routing Hub will return the merged output as response back to consumer

Figure 9-1 Extensibility - Example



> **Note:**
>
> Order of existing elements in custom template should be same as kernel template.

- XML merging attributes

9.1 XML merging attributes

This topic contains the following subtopics:

- Identity Matcher
- Skip Matcher

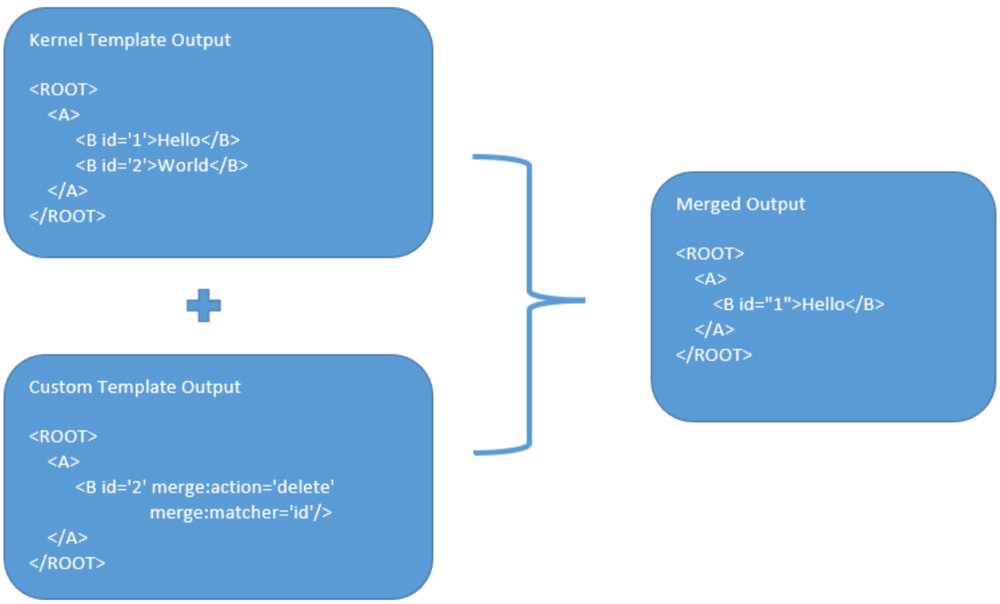
- Override Action
- Complete Action
- Replace Action
- Preserve Action
- Delete Action

9.1.1 Identity Matcher

Matcher attribute must be used when merge action has to be performed for specific element.

Syntax: `merge:matcher='<ATTRIBUTE_NAME>'`

Figure 9-2 Identity Matcher

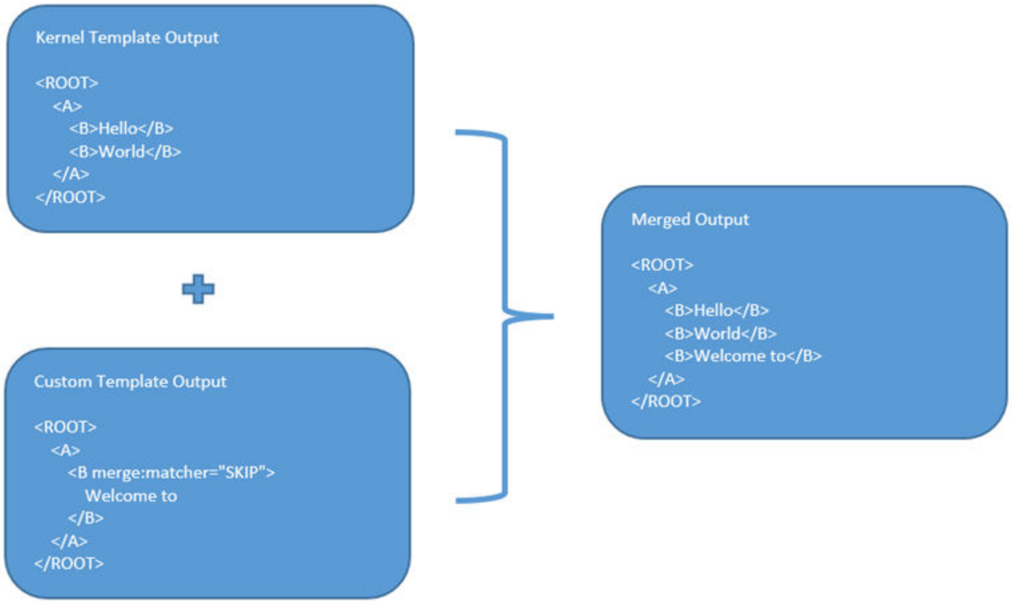


9.1.2 Skip Matcher

Skip matcher strategy is used to insert the elements forcefully without matching the original element and patch element.

Syntax: `merge:action='SKIP'`

Figure 9-3 Skip Matcher

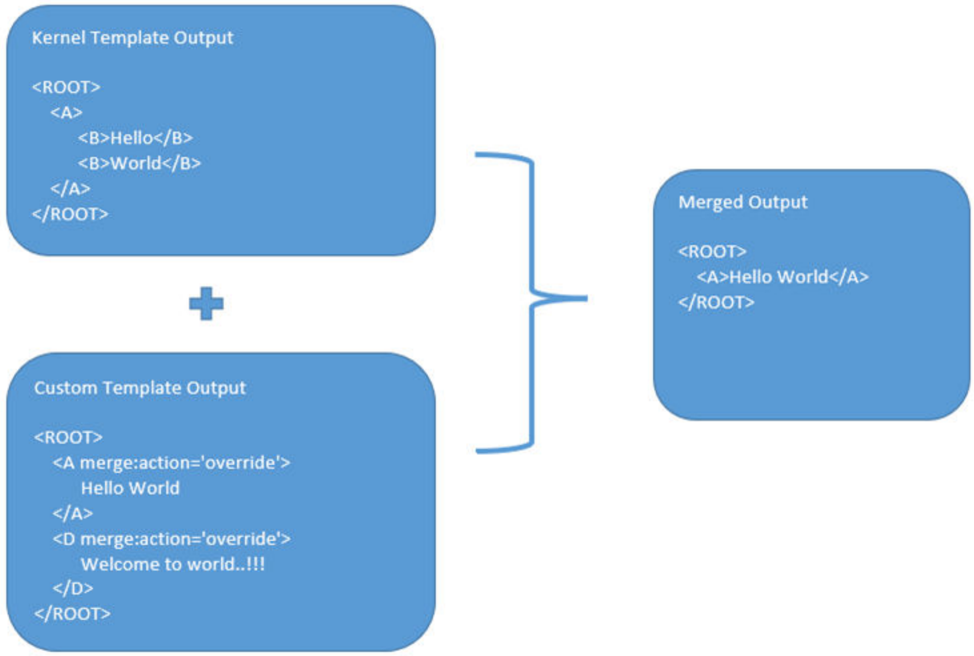


9.1.3 Override Action

Replaces the original element with the patch element only if it exists in kernel/mock template.

Syntax: `merge:action='override'`

Figure 9-4 Override Action

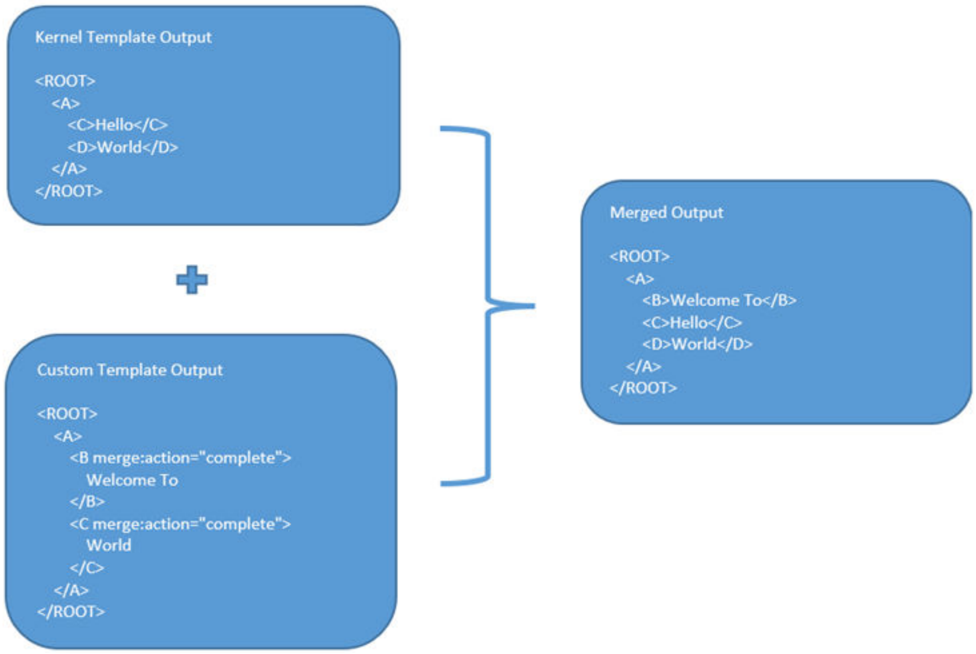


9.1.4 Complete Action

Copies the patch element only if it does not exist in kernel/mock template.

Syntax: `merge:action='complete'`

Figure 9-5 Complete Action

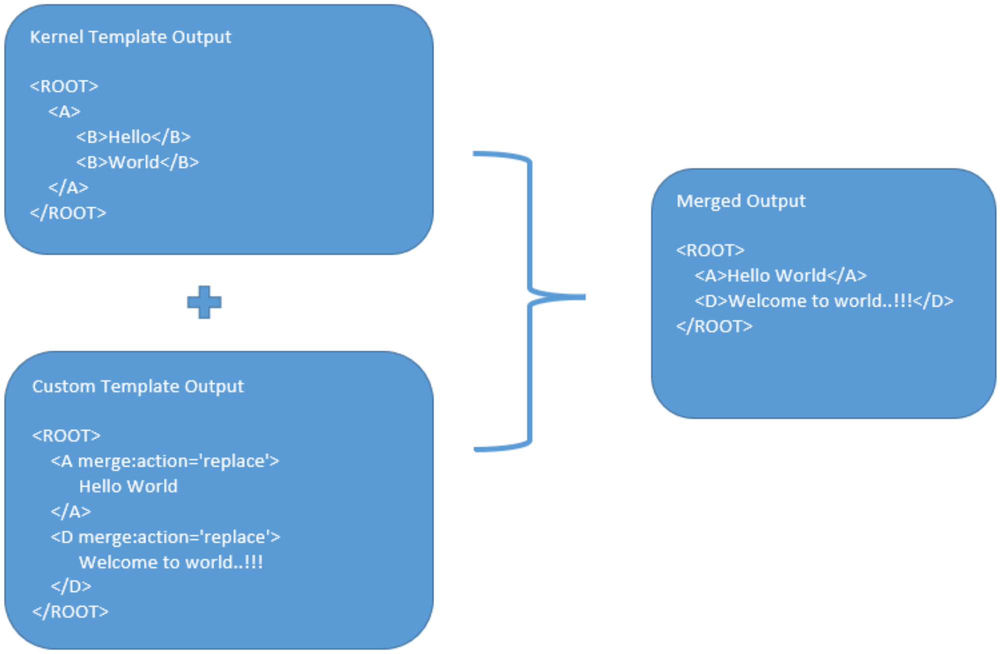


9.1.5 Replace Action

Replaces the original element with the patch element or creates the element if it does not exist in kernel/mock template.

Syntax: `merge:action='replace'`

Figure 9-6 Replace Action

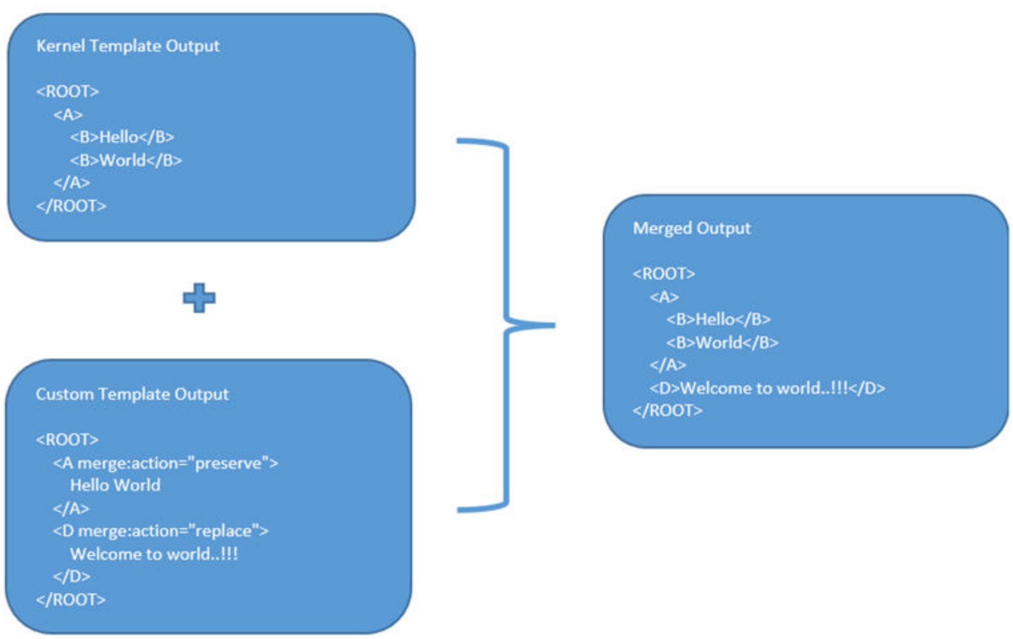


9.1.6 Preserve Action

No replace action is performed on the original element.

Syntax: `merge:action='preserve'`

Figure 9-7 Preserver Action

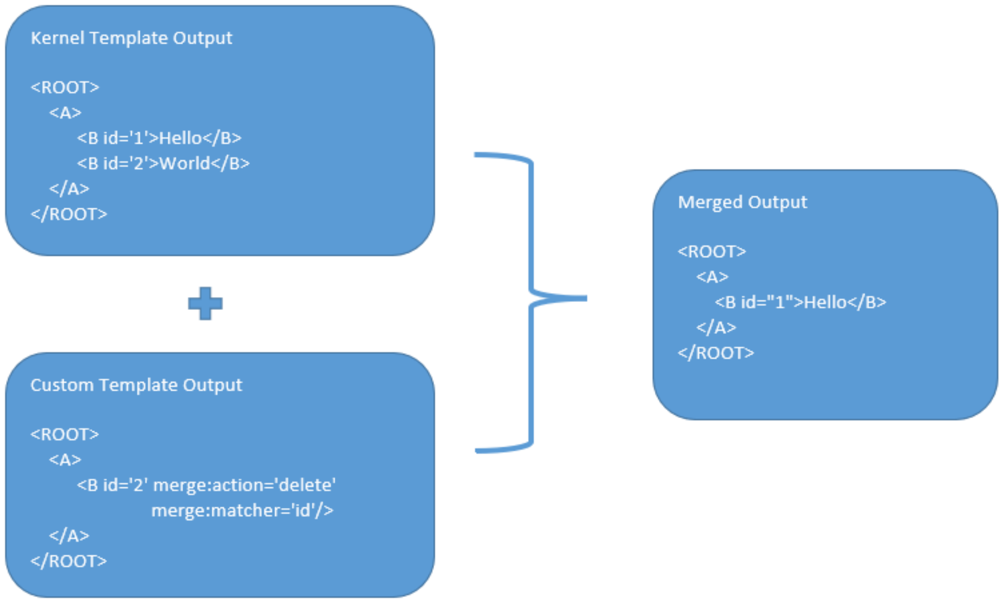


9.1.7 Delete Action

Deletes the original element.

Syntax: `merge:action='delete'`

Figure 9-8 Delete Action



10 Audit Purging / Archiving

Purging/Archiving of audit data is done on the basis of retention policy.

This process uses plato-batch-server for Job execution.

The following steps are required to schedule purging/archiving job (routingHubAuditRetentionJob) once cmc-obrh-services and plato-batch-server is UP and RUNNING:

1. On **Home** screen, click **Task Management**. Under **Task Management** menu, click **Configure Tasks**.
2. Select **Schedule** option.
3. Select **Task Name** as routingHubAuditRetentionJob and **Task Trigger Name** will be generated automatically.
4. Specify the CRON expression to daily EOD.

In order to resolve table space issue of Audit table (CMC_RH_AUDIT_EVENT_LOG), Database Management Team has to configure database job which should be triggered after routingHubAuditRetentionJob. This database job can be redefining the table (DBMS_REDEFINITION) after purging/archiving is done or other approach. So, the unused LOB segment space can be released. And in order to resolve table space issue of Audit history table (CMC_RH_AUDIT_EVENT_LOG_HISTORY), Database Management Team has to configure database job to truncate table periodically basis.

11 Multipart Request

This topic provides the sample template for the multipart request

Example 11-1   Multipart Request

```
[
    {
      "key": "file",
      "type": "FILE",
      "value": "$body.files.get(0).file"
    },
    {
      "key": "name",
      "type": "TEXT",
      "value": "$body.name.get(0).content"
    }
]
```

12 URL Encoded Request

This topic provides the sample template for url encoded request.

Example 12-1   URL Encoded Request

```
[
{
    "client_id": "am9obg",
    "client_secret": "am9obmRvZQ"
}
]
```

> **Note:**
>
> Body type should be RAW.

13
Configuration

This topic describes the systematic instructions to perform the configuration.

End-user can configure the properties w.r.t. monitoring, alert and export.

End-user can configure the same at System level and granular levels such as Consumer, Consumer Service and Routing.

The **Configuration** screen contains the following sections.

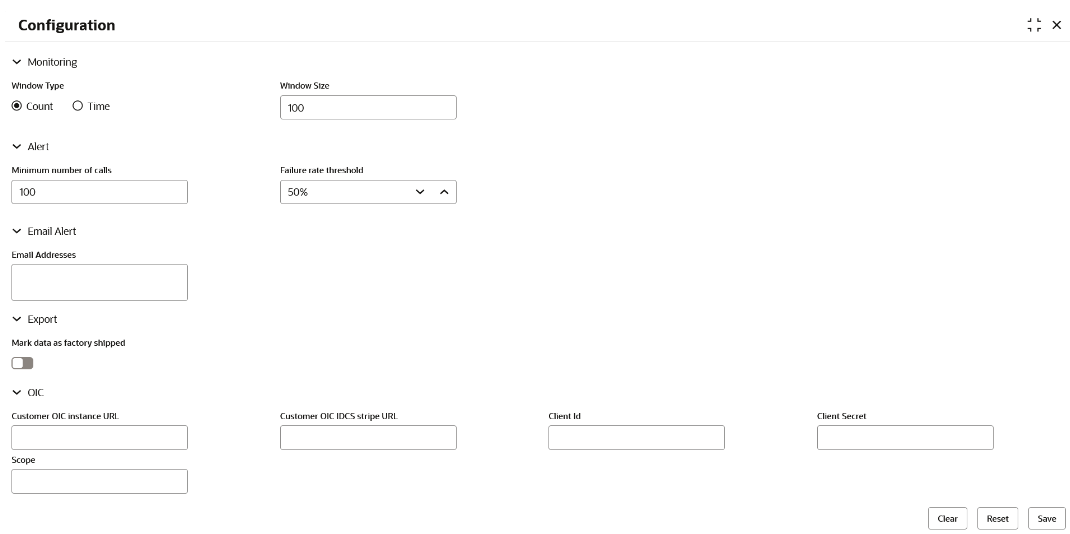
- **Monitoring** - It has the features required by the breaker to store and aggregate the result of calls.
- **Alert** - It has the features required for transitioning circuit breaker.
- **Email Alert** - It has the feature required for mail notification.
- **Export** - It has the properties that are required for exporting the configuration JSON and will be visible at system level only.

Specify **User ID** and **Password**, and login to **Home** screen.

1. On **Home** screen, click **Core Maintenance**. Under **Core Maintenance**, click **Routing Hub**.
2. Under **Routing Hub**, click **Configuration**.

   The **Configuration** screen displays.

   Figure 13-1 Configuration

3. Specify the fields on **Configuration** screen.

> **Note:**
>
> The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

Table 13-1 Configuration - Field Description

Field	Description
Window Type	Select the type of the window. The available options are: • **Count**: The count-based sliding window aggregates the outcome of the last N calls (**Window Size**). • **Time**: The time-based sliding window aggregates the outcome of the calls of the last N seconds (**Window Size**).
Window Size	Specify the window size to record the outcome of the calls when the circuit breaker is closed. • For **Count** window type, The window size is N calls. • For **Time** window type, The window size has N seconds.
Minimum number of calls	Specify the minimum number of calls. For example: If the minimum number of calls are 10, then at least 10 calls must be recorded before calculating the failure rate. If only nine calls are recorded, the circuit breaker is not transitioned to open even if all nine calls are failed.
Failure rate threshold	Specify the failure rate threshold in percentage. When the failure rate is equal or greater than the threshold, the circuit breaker transitions to open and starts short-circuiting calls.
Email Addresses	Specify the E-mail address. The user can use semi-colon to add more email addresses. Once the failure rate crosses the **Failure rate threshold**, a mail is sent to the end-user about the event.
Mark data as factory shipped	Select the toggle to mark the exported configuration JSON as factory shipped JSON. The end-user will not be able to modify or delete the certain data once imported. By default, the toggle is OFF.
Customer OIC instance URL	Specify the url of OIC instance.
Customer OIC IDCS stripe URL	Specify the striped url of IDCS.
Client Id	Specify the client identifier.
Client Secret	Specify the client secret.
Scope	Specify the intent of access.

Example:

Table 13-2 Configuration - Field Entry Values

Field	Entry Values
Window Type	Count
Window Size	20

Table 13-2 (Cont.) Configuration - Field Entry Values

Field	Entry Values
Minimum number of calls	10
Failure rate Threshold	50%

Configured properties will result as below:

After 10 (minimum number of calls) calls, routing would get shutdown if 50% (failure rate) of almost last 20 (window size) calls have failed. If the email address property is configured, then the end-user is notified as well.

4. Click **Clear** to clear all the specified details.
5. Click **Reset** to reset the details.
6. Click **Save** to save all the details.

14

Request Audit - Log

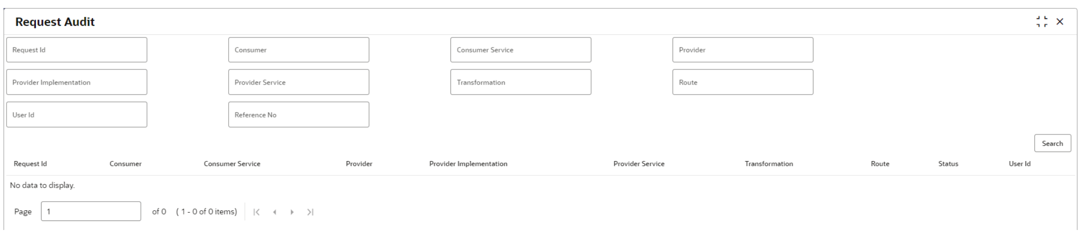
This topic describes the systematic instructions to check the audit log in Oracle Banking Routing Hub.

Specify **User ID** and **Password**, and login to **Home** screen.

1. On **Home** screen, click **Core Maintenance**. Under **Core Maintenance**, click **Routing Hub**.
2. Under **Routing Hub**, click **Request Audit**.

   The **Request Audit - Log** screen displays.

   Figure 14-1 Request Audit - log

3. Specify the fields on **Request Audit - log** screen.

> **Note:**
>
> The fields marked as **Required** are mandatory.

For more information on fields, refer to the field description table.

Table 14-1 Request Audit - log - Field Description

Field	Description
Request ID	Specify the request ID.
Consumer	Specify the consumer.
Consumer Service	Specify the consumer service.
Provider	Specify the provider.
Provider Implementation	Specify the provider implementation.
Provider Service	Specify the provider service.
Transformation	Specify the transformation name.
Route	Specify the route.
User ID	Specify the user ID.

Table 14-1 (Cont.) Request Audit - log - Field Description

Field	Description
Reference Number	Specify the reference number to track the requests audit. **Note:** To track by reference number, one has to pass rh-reference-no header in routing hub request

4. Click the **Search** button to fetch the request audit details.
5. Click on the **Request ID** to view the step by step execution of request audit details.

   The **Request Audit Details** screen displays.

Figure 14-2 Request Audit Details

For more information on fields, refer to the field description table.

Table 14-2 Request Audit Details - Field Description

Field	Description
Request ID	Displays the selected request ID.
OBRH Request	Displays the status of Routing Hub request.
Provider Request	Displays the status of provider request.
Provider Response	Displays the status of provider response.
OBRH Response	Displays the status of Routing Hub response.
Timestamp	Displays the date and time.
Message	Displays the message.

15 Monitoring Dashboard

Monitoring dashboard has been provided to System integrators and IT administrators to review the health of the integrations. It displays data using different type of widgets to help users to assess the performance of integrations and identify the areas that requires attention.

This dashboard requires ‘routingHubAuditSummaryJob' job to be executed periodically using plato-batch-server.

TBelow are steps to schedule the job once cmc-obrh-services and plato-batch-server is UP and RUNNING:

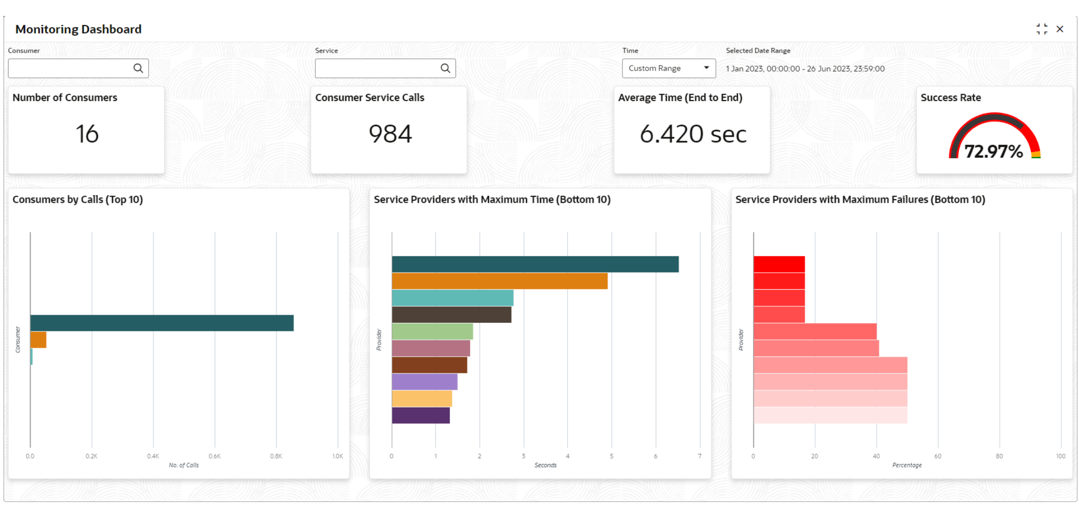
1. On **Home** screen, click **Task Management**. Under **Task Management** menu, click **Configure Tasks**.
2. Select **Schedule** option.
3. Select **Task Name** as routingHubAuditRetentionJob and **Task Trigger Name** will be generated automatically.
4. Specify the CRON expression to daily EOD.

To resolve table space issue of Audit summary table, (CMC_RH_AUDIT_SUMMARY), Database Management Team has to configure database job to truncate table periodically basis.

> **Note:**
>
> **Monitoring Dashboard** will also be not available if Audit logs are turned off.

1. On **Home** screen, click **Core Maintenance**. Under **Core Maintenance** menu, click **Routinh Hub**. Under **Routing Hub**, click **Monitoring Dashboard**

Figure 15-1 Monitoring Dashboard

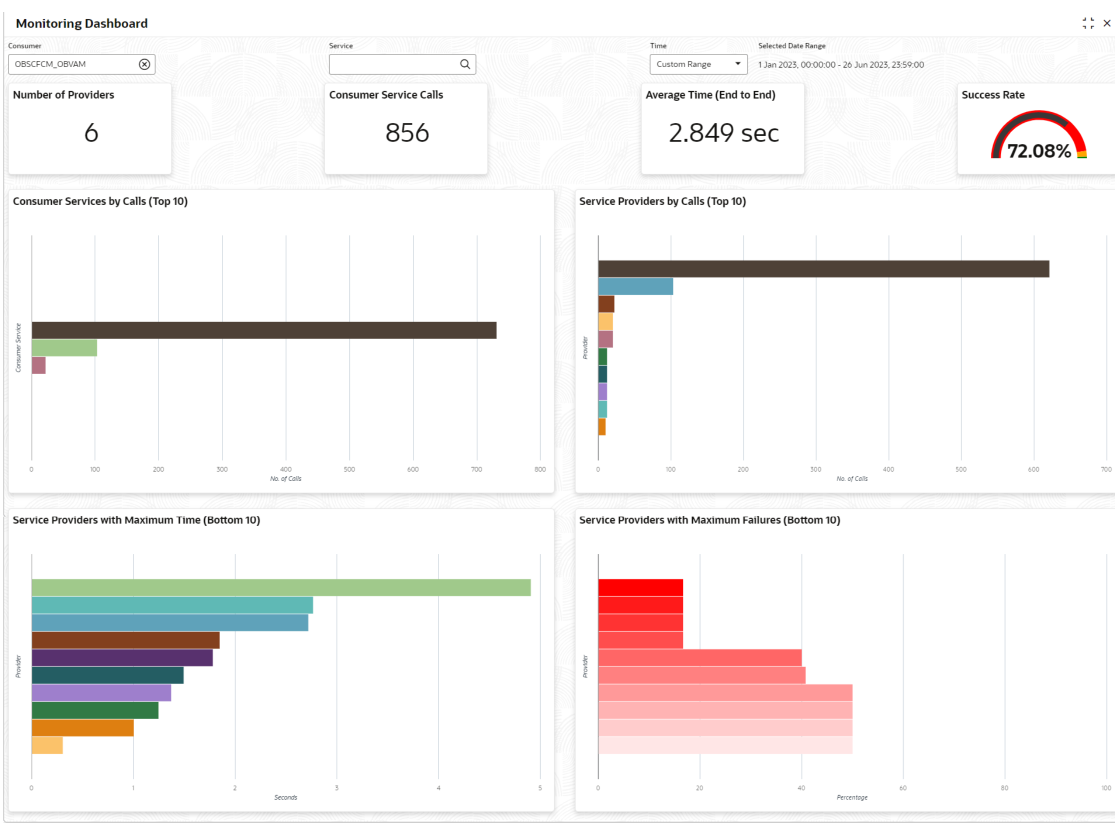


- **Number of Consumers:** This widget displays total number of consumers configured in the Oracle Banking Routing Hub.
- **Consumer Service Calls:** This widget displays total number of consumer services requested during chosen period.
- **Average Time (End to End):** This widget displays the average time (in seconds) taken to process successful requests, during chosen period.
- **Success Rate:** This widget provides an indicator of how many successful requests were made during chosen period.
- **Consumers by Calls (Top 10):** This widget provides a graphical display of the top 10 consumers based on requests they have made during chosen period. A link on the bar graph is provided to view further details of the Consumer.
- **Service Providers with Maximum Time (Bottom 10):** This widget provides a graphical display of bottom 10 providers based on the time taken to process requests, during s chosen period.
- **Service Providers with Maximum Failures (Bottom 10):** This widget provides a graphical display of bottom 10 providers based on failed requests, during s chosen period.

Consumer Page

The End-user can navigate to this page by either using the filter option provided on the landing page or by clicking on specific consumer service in "**Consumer Service by Calls (Top 10)**" chart.

Figure 15-2 Consumer Page



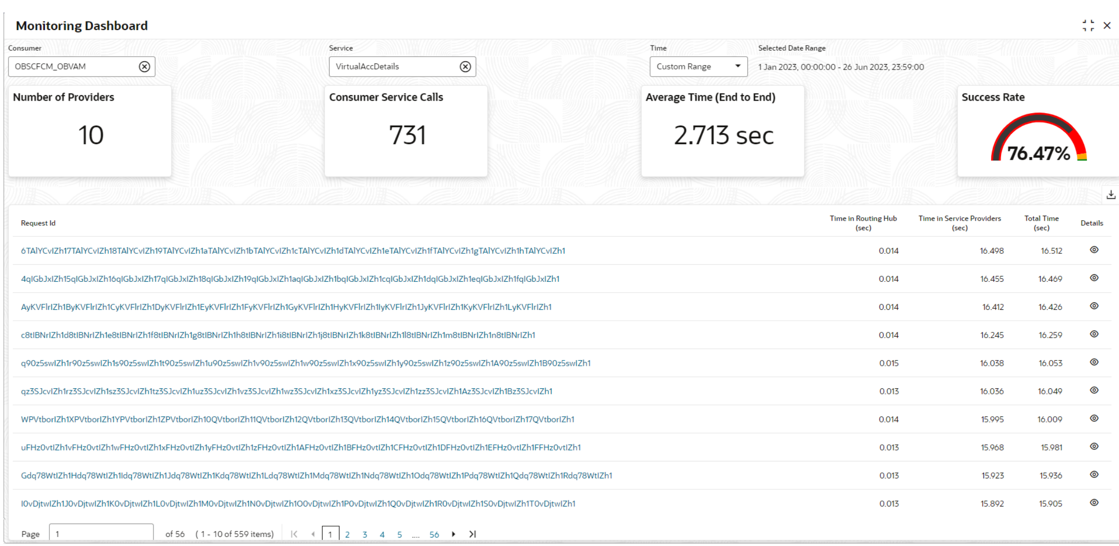
This page displays following information:

- **Number of Providers:** This widget displays the total number of service providers configured in Oracle Banking Routing Hub for the selected consumer.
- **Consumer Service Calls:** This widget displays total number of consumer services requested by the selected consumer during chosen period.
- **Average Time (End to End):** This widget displays the average time (in seconds) taken to process successful requests made by the selected consumer, during chosen period.
- **Success Rate:** This widget provides an indicator of how many successful requests were made by the selected consumer during chosen period.
- **Consumer Services by Calls (Top 10):** This widget provides a graphical display of the top 10 consumers Services during chosen period. A link on the bar graph is provided to view further details of the Consumer Service.
- **Service Providers by Calls (Top 10):** Shows top 10 service providers based on the maximum requests which are requested chosen period.
- **Service Providers with Maximum Time (Bottom 10):** Shows bottom 10 providers based on the maximum time taken to process successful requests which are requested during chosen period.
- **Service Providers with Maximum Failures (Bottom 10):** Shows bottom 10 providers based on the maximum number of failed requests which are requested during chosen period.

Consumer Service Page

The End-user can navigate to this page by either using the filter option provided on the landing page or by clicking on specific consumer service in "**Consumer Service by Calls (Top 10)**" chart.

Figure 15-3 Consumer Service Page



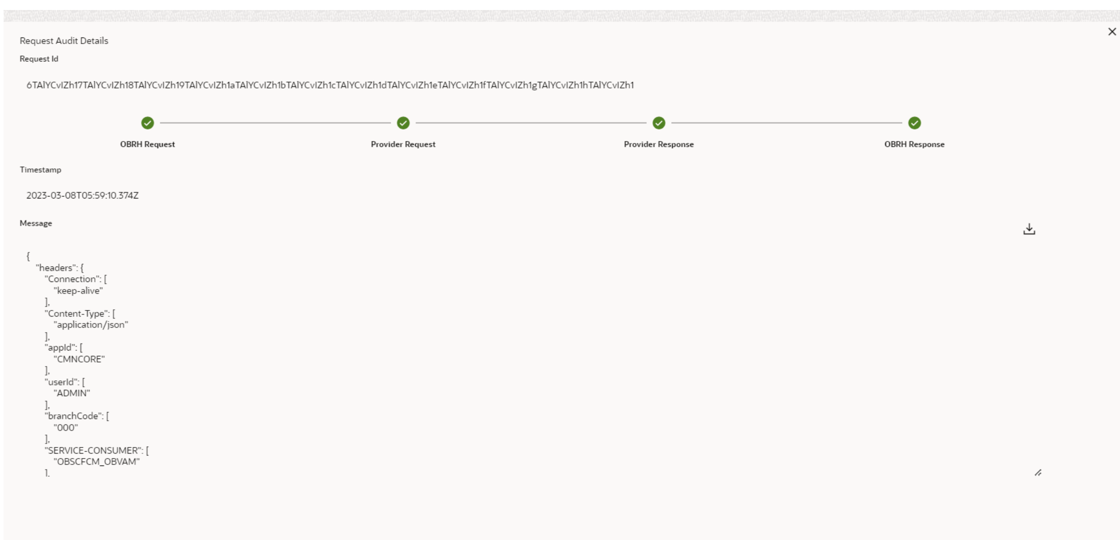
- **Number of Providers:** This widget displays total number of service providers to which this request is routed to complete the integration.
- **Consumer Service Calls:** This widget displays total number of consumer services made during chosen period.

- **Average Time (End to End):** This widget displays the average time taken to process successful requests made during chosen period.
- **Success Rate:** Shows the percentage of successful requests which are made during chosen period.
- **Request Details:** The table displays the list of requests which are requested during chosen period. Following are the details which are provided for each request.

Component Name	Component Type
Request Id	This is system generated reference number for each request. Click on the Request Id displays audit log information of the request.
Time in Routing Hub (Sec)	This field displays the time taken by Routing Hub (in seconds) to route the request between Consumer Service and Providers.
Time in Service Providers (Sec)	This field displays the total time taken by Service provides (in seconds) to process the request.
Total Time (Sec)	This field displays the total time to process the request
Provider Service	Text box
Details	Displays the tabular view of the time taken by individual providers (in case of chaining of the request)

End-user can view request details by clicking on Request Id.

Figure 15-4 Request Audit Details



Component Name	Comments
Number of Providers	Shows total number of service providers.
Consumer Service Calls	Shows total number of consumer services requested during selected time.
Average Time (End to End)	Shows the average time taken to process successful requests which are requested during selected time.

Component Name	Comments
Success Rate	Shows the percentage of successful requests which are requested during selected time.
Request Audit	Shows list of requests which are requested during selected time.

16 Transformation Type

This topic provides the information about the transformation types.

Velocity

Velocity is a Java-based template engine. It is used to generate XML files, SQL, PostScript, and most other text-based formats.

> **Note:**
>
> In Routing Hub, velocity is used to generate JSON and XML files.

- Using **$body**, user can access request/response body.
  Syntax: $body.fieldName

  Example: $body.branchCode
- Using **$headers**, user can access request/response headers.
  Syntax: $headers["fieldName"][0]

  Example: $headers["branchCode"][0]
- Using **$bodyAsString**, user can access response body as string.
  Syntax: $bodyAsString
- Below are some available extension methods:
  - Date Conversion
    Syntax: $dateUtil.convert(inputDate, fromPattern, toPattern)

    Parameters:

    * inputDate - String
    * fromPattern - String
    * toPattern - String

    Returns: String

    Refer to https://docs.oracle.com/javase/8/docs/api/java/text/SimpleDateFormat.html for different patterns
  - Default Value
    Syntax: $custom.defaultValue(inputValue, defaultValue)

    Parameters:

    * inputValue - Object
    * defaultValue - String

    Returns: Object
  - Null Check
    Syntax: $custom.isNull(inputValue)

Parameters:

* inputValue - Object

Returns: Boolean

- Random Number
  Syntax: $mathUtil.getRandom()

  Returns: Object of Random class (java.util.Random)

- Xml Tool
  Syntax: $xml.methodName()

  Refer to https://velocity.apache.org/tools/3.1/apidocs/org/apache/velocity/tools/generic/XmlTool.html

- Date Tool
  Syntax: $date.methodName()

  Refer to https://velocity.apache.org/tools/3.1/apidocs/org/apache/velocity/tools/generic/DateTool.html

- Json Tool
  Syntax: $json.methodName()

  Refer to https://velocity.apache.org/tools/3.1/apidocs/org/apache/velocity/tools/generic/JsonTool.html

- Math Tool
  Syntax: $math.methodName()

  Refer to https://velocity.apache.org/tools/3.1/apidocs/org/apache/velocity/tools/generic/MathTool.html

- Number Tool
  Syntax: $number.methodName()

  Refer to https://velocity.apache.org/tools/3.1/apidocs/org/apache/velocity/tools/generic/NumberTool.html

- Escape Tool
  Syntax: $esc.methodName()

  Refer to https://velocity.apache.org/tools/3.1/apidocs/org/apache/velocity/tools/generic/EscapeTool.html

- Serialization of object into its equivalent Json representation
  Syntax: $custom.toJson(src)

  Parameters:

  * src - Object

  Returns: String

- Get additional field's value based on fieldname
  Syntax: $custom.getFieldValueById(jsonString, fieldname)

  Parameters:

  * jsonString – String
  * fieldname - String

  Returns: String

- Get list of additional fields based on fieldname prefix
  Syntax: $custom.getAdditionalFieldSetByType(jsonString,prefixval)

Parameters:

* jsonString - String
* prefixval - String

Returns: String

– This method is for parsing XML string
Syntax: $custom.parseXml(xmlString)

Parameters:

* xmlString - String

Returns: Object

- If issue occurred with hyphen in velocity template of Request or Response Transformation, then use get method.
Example:

```
<FCUBS_BODY>
   <Customer-IO>
     <CUSTNO>003942</CUSTNO>
   </Customer-IO>
</FCUBS_BODY>
```

If `$in.FCUBS_BODY.Customer-IO.CUSTNO` does not work ,
use `$in.FCUBS_BODY.get("Customer-IO").CUSTNO` to get customer number.

XSLT

XSLT is a language for transforming XML documents into other XML documents, or other formats such as HTML for web pages, plain text or XSL formatting objects, which may subsequently be converted to other formats, such as PDF, PostScript and PNG.

> **Note:**
>
> In Routing Hub, XSLT is used to transform arbitrary XML to JSON.

JSLT

JSLT is a complete query and transformation language for JSON.

A

Functional Activity Codes

Table A-1 List of Functional Activity Codes

Screen Name	Functional Activity Codes	Action	Description
Routing Hub	CMC_FA_RH_APPLICATION	VIEW	Service Consumers UI in Routing Hub
Routing Hub	CMC_FA_RH_AUDIT_LOG	CREATE	Log audit information in Routing Hub
Routing Hub	CMC_FA_RH_AUDIT_SUMMARY	GET	Audit Summary
Routing Hub	CMC_FA_RH_AUDIT_SUMMARY_DATA	GET	Audit Summary Data
Routing Hub	CMC_FA_RH_CLEAR_SOAP_CLIENT_CACHE	CLEAR	Clears Soap Client Cache in Routing Hub
Routing Hub	CMC_FA_RH_CONFIG	VIEW	Configuration UI in Routing Hub
Routing Hub	CMC_FA_RH_CONFIG_CREATE	CREATE	Creates configuration
Routing Hub	CMC_FA_RH_CONFIG_DELETE	DELETE	Deletes configuration
Routing Hub	CMC_FA_RH_CONFIG_GET	GET	Fetches configuration
Routing Hub	CMC_FA_RH_CONFIG_MODIFY	MODIFY	Updates configuration
Routing Hub	CMC_FA_RH_CONSUMER_QUEUE_CREATE	CREATE	Saves new Consumer Queue Mapping
Routing Hub	CMC_FA_RH_CONSUMER_QUEUE_DELETE	DELETE	Deletes specific Consumer Queue Mapping
Routing Hub	CMC_FA_RH_CONSUMER_QUEUE_GETALL	GET	Fetches all Consumer Queue Mappings
Routing Hub	CMC_FA_RH_CONSUMER_QUEUE_GETBYID	GET	Fetches specific Consumer Queue Mapping
Routing Hub	CMC_FA_RH_CONSUMER_QUEUE_MODIFY	MODIFY	Updates specific Consumer Queue Mapping
Routing Hub	CMC_FA_RH_DASHBOARD	VIEW	Monitoring Dashboard UI
Routing Hub	CMC_FA_RH_DISPATCH_AUDIT_GETALL	GET	Fetches routing hub requests from audit log
Routing Hub	CMC_FA_RH_DISPATCH_AUDIT_LOG	VIEW	Request Audit UI in Routing Hub
Routing Hub	CMC_FA_RH_ROUTE_DISPATCH	INTEGRATION CALL	Synchronous/Asynchronous integration call
Routing Hub	CMC_FA_RH_ROUTE_DISPATCH_RESPONSE	GET	Fetches provider response of asynchronous routing hub request
Routing Hub	CMC_FA_RH_SERVICECONSUMER_CREATE	CREATE	Creates consumer

Table A-1 (Cont.) List of Functional Activity Codes

Screen Name	Functional Activity Codes	Action	Description
Routing Hub	CMC_FA_RH_SERVICECONSUMER_DELETE	DELETE	Deletes consumer
Routing Hub	CMC_FA_RH_SERVICECONSUMER_ENV_VARIABLE_EXPORT	EXPORT	Exports environment variables from Routing Hub Maintenance
Routing Hub	CMC_FA_RH_SERVICECONSUMER_ENV_VARIABLE_IMPORT	IMPORT	Imports environment variables
Routing Hub	CMC_FA_RH_SERVICECONSUMER_EXPORT	EXPORT	Exports consumer
Routing Hub	CMC_FA_RH_SERVICECONSUMER_GETALL	GET	Fetches all consumers
Routing Hub	CMC_FA_RH_SERVICECONSUMER_GETBYID	GET	Fetches specific consumer
Routing Hub	CMC_FA_RH_SERVICECONSUMER_IMPORT	IMPORT	Imports consumer
Routing Hub	CMC_FA_RH_SERVICECONSUMER_MODIFY	MODIFY	Updates consumer
Routing Hub	CMC_FA_RH_SERVICECONSUMER_PROCESSJSON	GET	Extracts configuration from configuration file
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICEROUTING_CREATE	CREATE	Creates route
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICEROUTING_DELETE	DELETE	Deletes route
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICEROUTING_GETALL	GET	Fetches all routes
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICEROUTING_GETBYID	GET	Fetches specific route
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICEROUTING_MODIFY	MODIFY	Updates route
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICETRANSFORMATION_CREATE	CREATE	Creates transformation
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICETRANSFORMATION_DELETE	DELETE	Deletes transformation
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICETRANSFORMATION_EXPORT	EXPORT	Exports transformation
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICETRANSFORMATION_GETALL	GET	Fetches all transformations
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICETRANSFORMATION_GETBYID	GET	Fetches transformation
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICETRANSFORMATION_IMPORT	IMPORT	Imports transformation
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICETRANSFORMATION_MODIFY	MODIFY	Updates transformation
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICE_CREATE	CREATE	Creates service
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICE_DELETE	DELETE	Deletes service
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICE_EXPORT	EXPORT	Exports service

Table A-1 (Cont.) List of Functional Activity Codes

Screen Name	Functional Activity Codes	Action	Description
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICE_GETALL	GET	Fetches all services
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICE_GETBYID	GET	Fetches specific service
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICE_IMPORT	IMPORT	Imports service
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICE_MODIFY	MODIFY	Updates service
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_CREATE	CREATE	Creates provider
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_DELETE	DELETE	Deletes provider
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_EXPORT	EXPORT	Exports provider
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_GENERATEREQUEST	GET	Extracts provider service's request definition
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_GETALL	GET	Fetches all providers
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_GETBYID	GET	Fetches provider
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_IMPL_CREATE	CREATE	Creates implementation
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_IMPL_DELETE	DELETE	Deletes implementation
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_IMPL_EXPORT	EXPORT	Exports implementation
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_IMPL_GENERATEREQUEST	GET	Extracts implementation service's request definition
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_IMPL_GETALL	GET	Fetches all implementations of specific provider
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_IMPL_GETBYID	GET	Fetches specific implementation of specific provider
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_IMPL_IMPORT	IMPORT	Imports implementation
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_IMPL_MODIFY	MODIFY	Updates implementation
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_IMPORT	IMPORT	Imports provider
Routing Hub	CMC_FA_RH_SERVICEPROVIDER_MODIFY	MODIFY	Updates provider
Routing Hub	CMC_FA_RH_TEMPLATE_EVALUATION	GET	Evaluates transformation template

Index