Oracle® Banking Microservices Architecture

Routing Hub Configuration User Guide

Release 14.8.0.0.0
G29394-03
April 2025

Oracle Banking Microservices Architecture Routing Hub Configuration User Guide, Release 14.8.0.0.0

G29394-03

Contents

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.

Critical Patches

Oracle advises customers to get all their security vulnerability information from the Oracle Critical Patch Update Advisory, which is available at Critical Patches, Security Alerts and Bulletins. All critical patches should be applied in a timely manner to make sure effective security, as strongly recommended by Oracle Software Security Assurance.

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.
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

Table 3 (Cont.) Symbols and Icons - Common

Symbol/Icon	Function
	Navigate to the next record
	Refresh
	Click this icon to delete a row, which is already added.
	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.

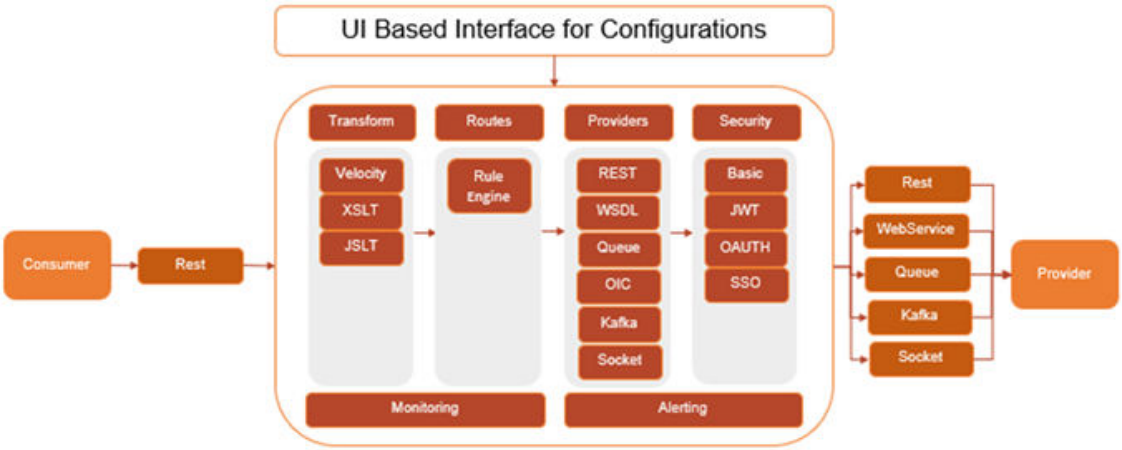
- **Servicing Providers or Product Processors**: The consumer application requests data from the products when required, or a consumer application initiates a transaction for the products to post.
- **Name of the Service**: Logical name of the service example: The service provider's product allows us 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, Kafka, 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 solution that utilizes the Oracle Banking Routing Hub API for integration purposes. Analyze the Oracle Banking Routing Hub and assess the destination product processor. Convert the data into the necessary format for the destination product processor to handle a specific 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

New Service Consumer

The users can create **Service Consumers** manually.

3. Click **New**.

   The **New Service Consumer** screen displays.

Figure 2-2 New Service Consumer

4. On **New Service Consumer** screen, specify the fields.

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

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

Table 2-1 New Service Consumer - Field Description

Field	Description
Consumer Name	Specify a unique service consumer name. **Note:** • Enter 0 to maximum of 255 characters. • No numeric value at beginning and no space allowed.
Consumer Description	Specify the description of the consumer name. **Note:** • Enter 0 to maximum of 100 characters. • No space allowed at beginning or ending of the characters.

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

Field	Description
Request Audit Type	Select the Audit type from the drop-down list. 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

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

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.

6. Click **Import**.

   The **Import Service Consumer** screen displays.

Figure 2-3 Import Service Consumer - Basic Details

7. Specify the fields on **Import Service Consumer - Basic Details** screen.

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

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

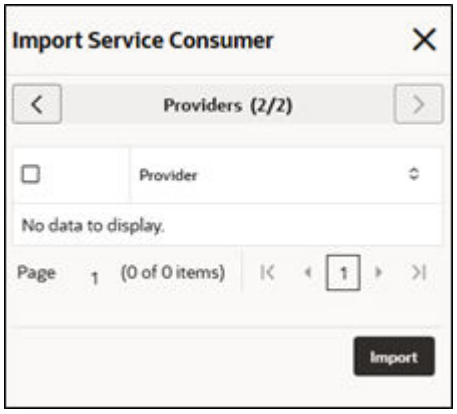
Table 2-2 Import Service Consumer - Basic Details - Field Description

Field	Description
File	Click **Select** to select the file. **Note:** Only one file can be selected, and it must be in JSON or ZIP format.
Name	Specify the name of the service consumer. **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.
Providers	Displays the list of service providers names that are present in JSON file only.

8. Click **Next** on the **Basic Details** screen.

   The **Import Service Consumer - Providers** screen displays.

Figure 2-4 Import Service Consumer - Providers



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

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

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

Table 2-3 Import Service Consumer - Providers - Field Description

Field	Description
Providers	Displays the list of service providers names that are present in JSON file only.

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

View / Edit Service Consumer
The user can view or modify consumer details.

11. On the **Service Consumer** tile, click **View More** button and then click **Edit Service Provider**.

    The **Edit Service Consumer** screen displays.

Figure 2-5 Edit Service Consumer

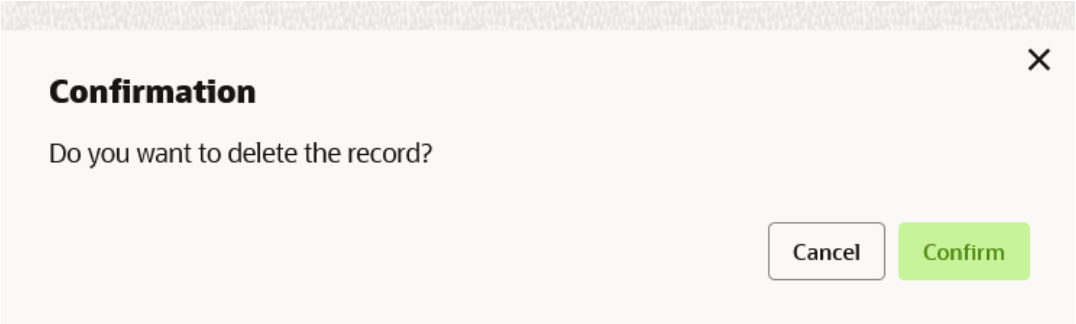
12. Click **Save** to save the modified consumer details.

Delete Service Consumer
The user can delete the Service Consumer.

13. On the **Service Consumer** tile, click **Delete** icon.

    The **Confirmation** screen displays.

Figure 2-6 Confirmation - Delete



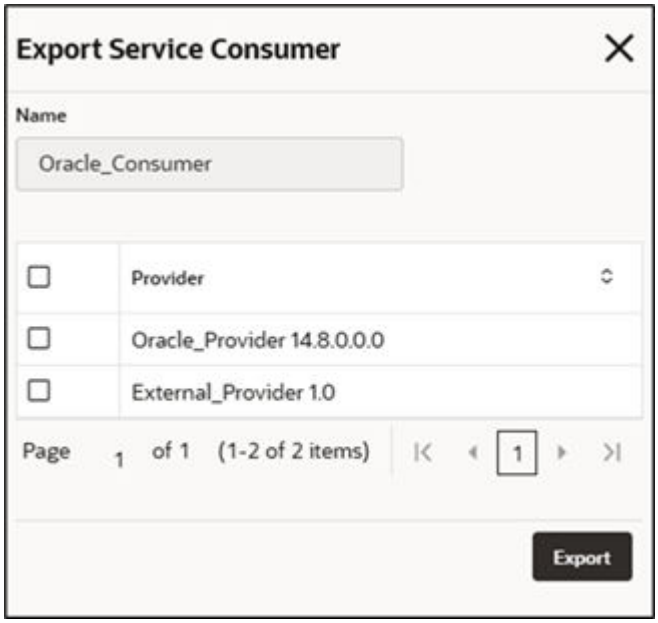
14. Click **Confirm** to delete the service consumer.

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

15. On **Service Consumer** tile, click **Operation Menu** (3 dot icon) and then click **Export**.

    The **Export Service Consumer** screen displays.

Figure 2-7 Export Service Consumer



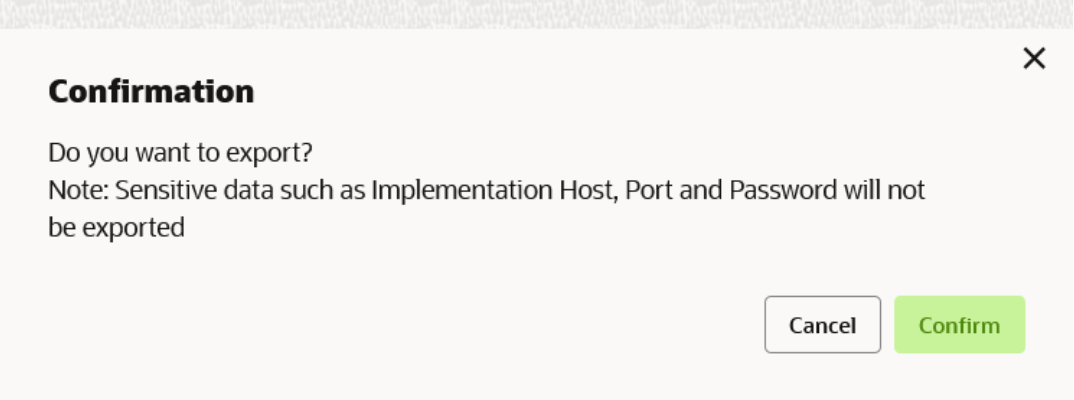
Note:

- 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

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

    The **Confirmation** screen displays.

Figure 2-8 Confirmation - Export



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

Configuration Export

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

    The **Configuration** screen is displayed.

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

Request Audit

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

    The **Request Audit** screen is displayed.

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

3
Environment Variables

This topic describes the systematic instructions to configure the environment variables consumers.

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

The user needs to define a set of variables that will be accessible across the particular configuration of the consumer. The syntax for accessing environment variables is below: $env.Environment_Group_Name.Environment_Variable_Name

For example, $env.COMMON.BRANCH_CODE

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

   The **Environment Variables** screen is displayed.

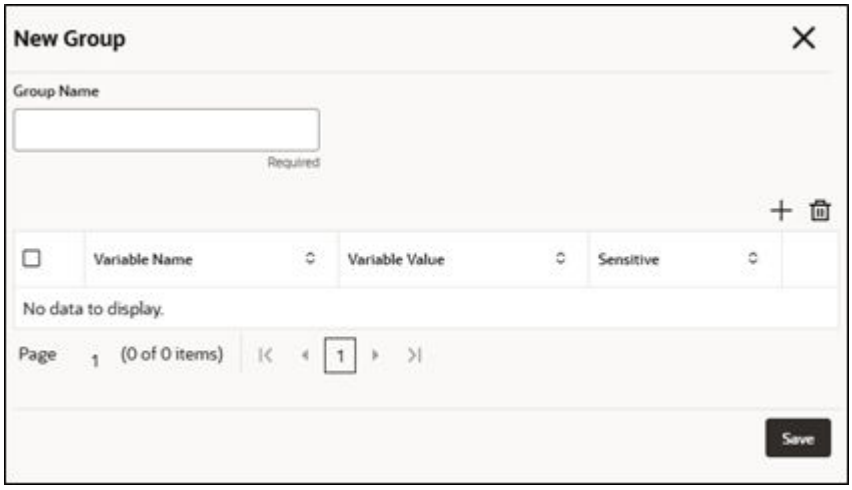
Figure 3-1 Environment Variables

New Group
Users can create multiple groups and variables.

2. Click **New Group**.

   The **New Group** screen is displayed.

Figure 3-2 New Group



3. On **New Group** screen, specify the fields.

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

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

Table 3-1 New Group - Field Description

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

Table 3-1 (Cont.) New Group - Field Description

Field	Description
Sensitive	With this flag, user can mark the variables which are sensitive in nature. So, The values will not be shown as plain text in routing hub configuration.

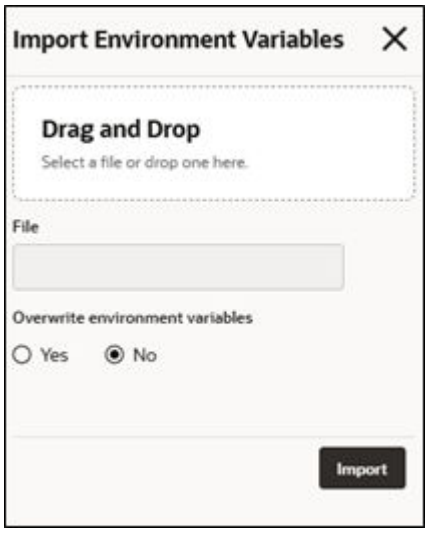
4. Click **Save** to save the details.

Import Group
The user can import environment variables.

5. Click **Import Environment Variables**.

   The **Import Environment Variables** screen displays.

Figure 3-3 Import Environment Variables



6. Specify the fields on **Import Environment Variables** screen.

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

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

Table 3-2 Import Environment Variables - Field Description

Field	Description
File	Select the file using **Select**. **Note:** Allows only to select one file and accepts JSON and ZIP file.
Overwrite environment variables	Select the respective radio button to specify if the environment variables (from the JSON file) should replace the current environment variables. The options available are: • **Yes** - This option overwrites the environment variables. • **No** - This option retains the existing environment variables.

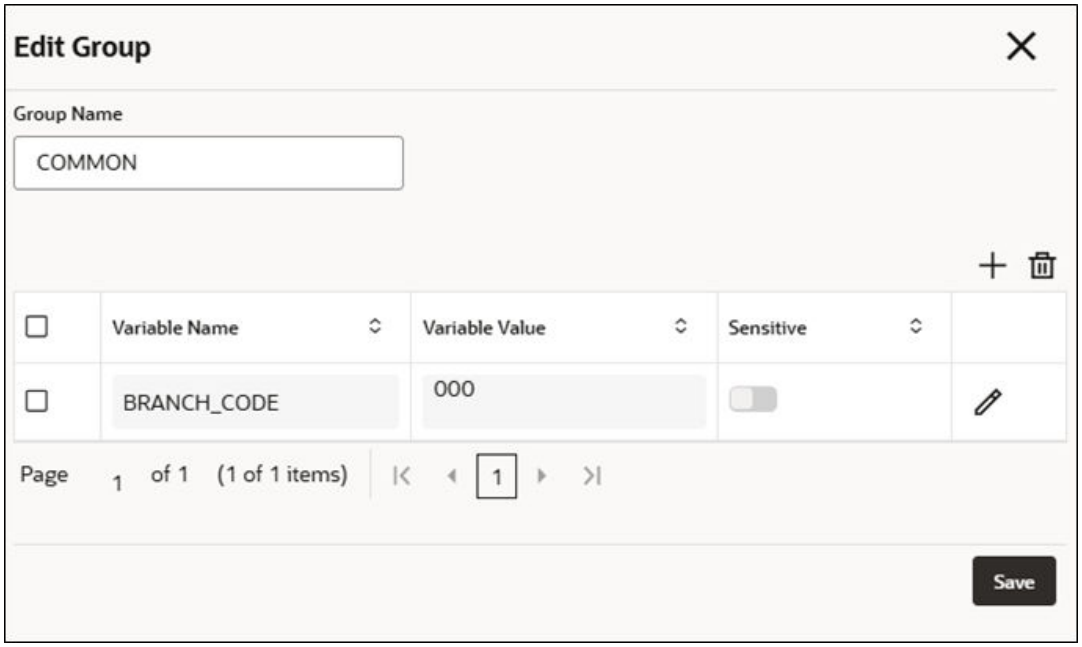
7. Click **Import** to import the selected environment variable file.

View / Edit Group
The user can view or modify environment variables.

8. On the **Group** tile, click **Edit Group**.

   The **Edit Group** screen is displayed.

Figure 3-4 Edit Group



9. Click **Save** to save the modified environment variable details.

Delete Group
The user can delete the environment group..

10. On the **Group** tile, click **Delete** icon.

Export Group

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

11. On **Environment Variables** screen, click **Export Group**.

    The **Confirmation - Export** screen is displayed.

Figure 3-5 Confirmation - Export

12. Click **Confirm** to export the environment variables in JSON file.

4
Service Providers

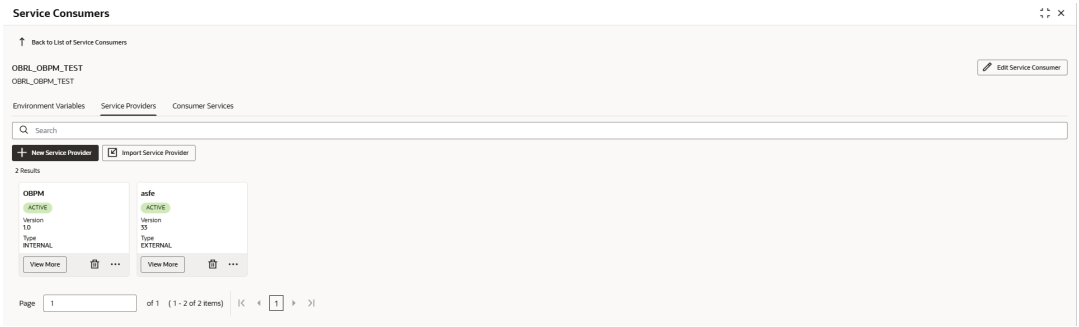
This topic describes the systematic instructions to configure the service providers.

Service Providers are systems designed to handle requests sent by the Oracle Banking Routing Hub for service consumers. They include information about destination integration.

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

   The **Service Providers** screen is displayed.

Figure 4-1 Service Providers

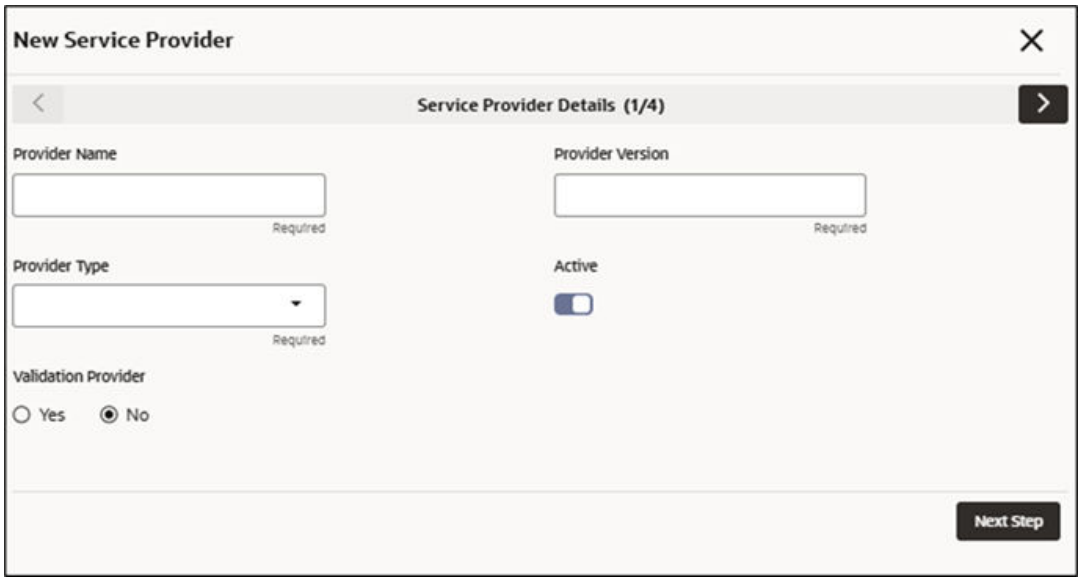


New Service Provider
The user can create Service Provider manually.

2. Click **New**.

   The **New Service Provider** screen is displayed.

Figure 4-2 New Service Provider - Service Provider Details



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

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

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

Table 4-1 New Service Provider - Service Provider Details - Field Description

Field	Description
Provider Name	Specify the name of the service provider. **Note:** • Enter 0 to maximum of 255 characters. • No numeric value at beginning and no space allowed.
Provider Version	Specify the provider version. **Note:** • Enter 0 to maximum of 255 characters. • Only numeric or decimal values are allowed.
Provider 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.
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 is used to mark the service provider to also act as a validator for validating the requests before sending it for further processing.

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. Click **Next Step**.

The **New Service Provider - Headers**

Figure 4-3 New Service Provider - Headers

5. Specify the fields on **New Service Provider - Headers** screen.

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

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

Table 4-2 New Service Provider - Headers - Field Description

Field	Description
Name	Specify the name of the header.
Value	Specify the value of the header. Value can be hardcoded or velocity template.

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.

  Context path can be modified for existing WSDL operations.

  > **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.

Existing REST endpoints can also be modified or deleted.

> **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.

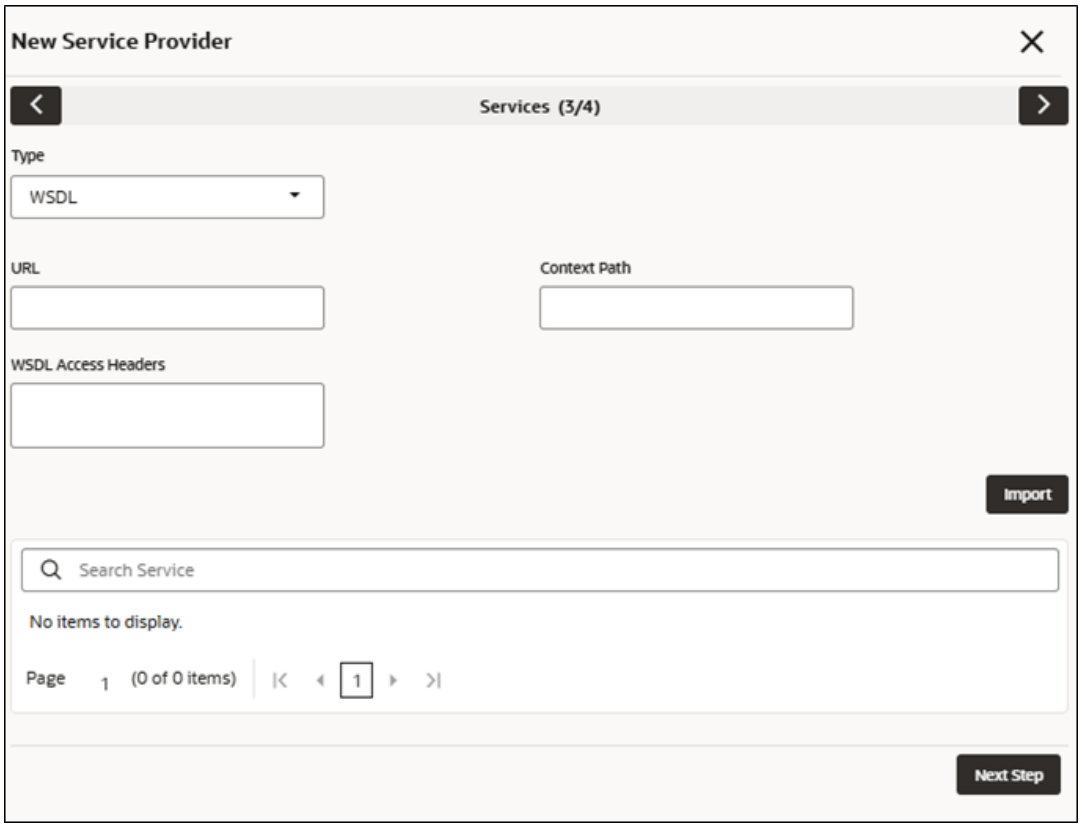
- **Others**:
  Others option is selected for adding REST API details manually when provider does not have swagger file.

> **Note:**
>
> If there is a change in existing endpoint, then the same endpoint details can be modified using edit option.

6. Click **Next Step**.

   The **New Service Provider - Services**

Figure 4-4 New Service Provider - Services



7. On **New Service Provider - Services** screen, specify the fields.

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

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

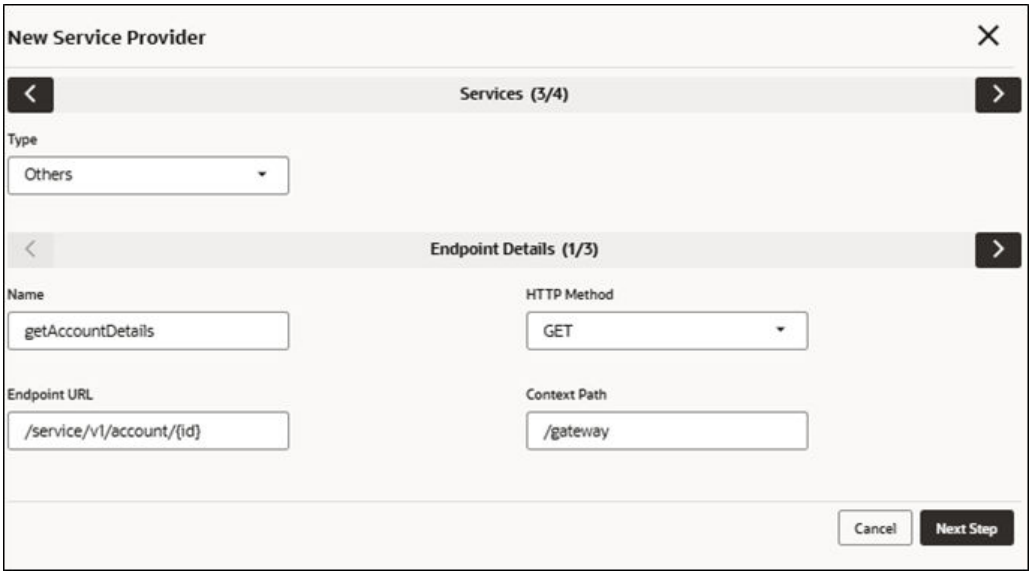
Table 4-3 New Service Provider - Services - Field Description

Field	Description
Type	Select the service type from drop-down list. The available options are: • **WSDL** • **SWAGGER** • **OTHERS**
URL	Specify the service URL of the file location. **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
WSDL Access Headers	Specify the headers required for accessing / reading WSDLs.
Import	Click **Import** to extract the service information from URL. **Note:** This field appears only if the **Type** is selected as **WSDL** and **SWAGGER**.

a. On **New Service Provider** screen, for adding REST endpoint details manually, select the **Type** as **Others** to define the endpoint details.

   The **Endpoint Details** screen is displayed.

Figure 4-5 Endpoint Details



b. Specify the fields on **Endpoint Details** screen.

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

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

Table 4-4 Endpoint Details - Field Description

Field	Description
Name	Specify the name of the operation. **Note:** This field appears only if the **Type** is selected as **Others**.

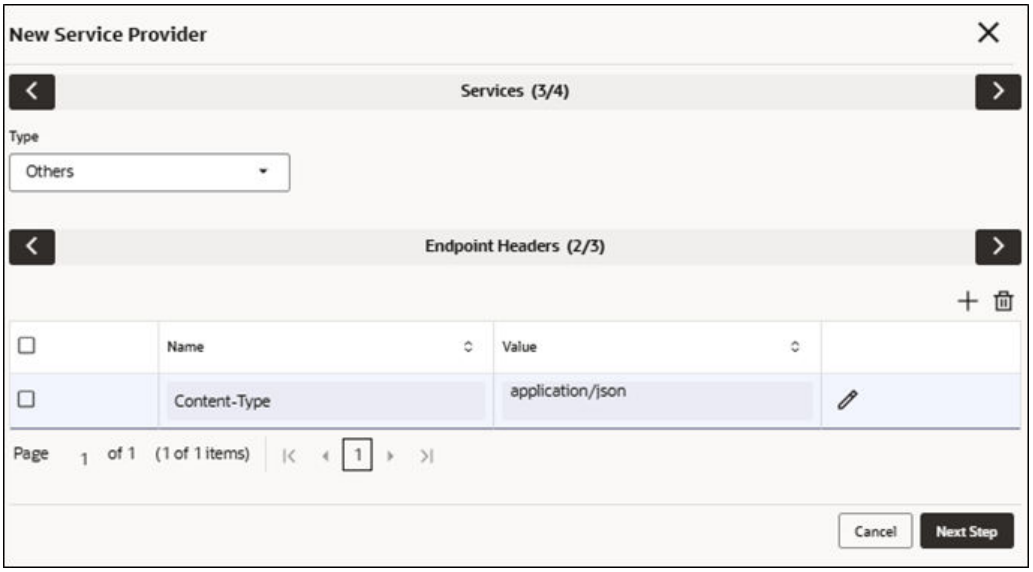
Table 4-4 (Cont.) Endpoint Details - Field Description

Field	Description
HTTP Method	Select the HTTP method from the drop-down list. The available options are: • **GET** • **POST** • **PUT** • **PATCH** • **DELETE** **Note:** This field appears only if the **Type** is selected as **Others**.
Endpoint URL	Specify the endpoint URL for the operation. **Note:** This field appears only if the **Type** is selected as **Others**.
Content Path	Context path of below formatted URL http://host:port/context-path/endpoint

c. Click **Next Step**.

The **Endpoint Headers** screen is displayed.

Figure 4-6 Endpoint Headers



d. Specify the fields on **Endpoint Headers** screen.

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

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

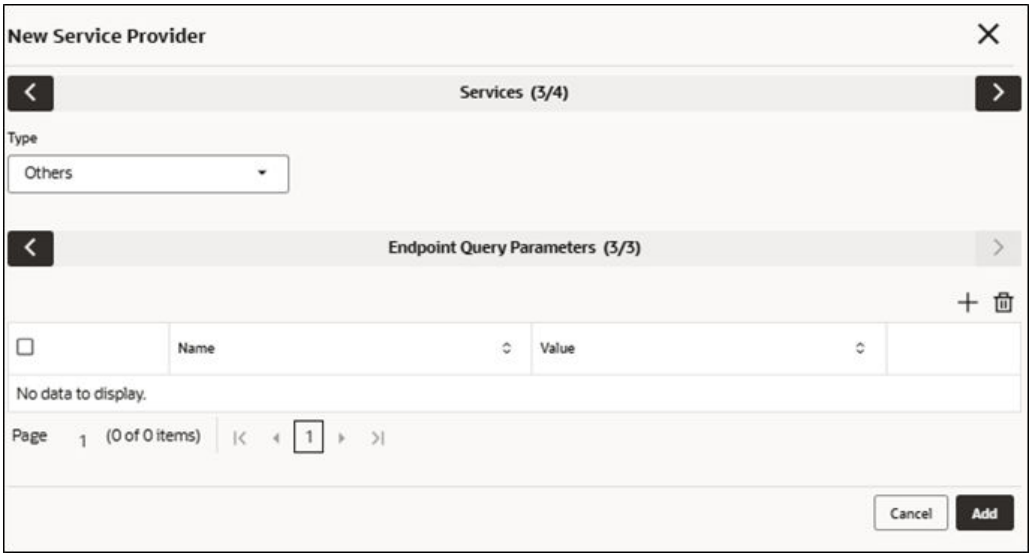
Table 4-5 Endpoint Headers - Field Description

Field	Description
Name	Specify the name of the header. **Note:** This field appears only if the **Type** is selected as **Others**.
Value	Specify the value of the header. Value can be hardcoded or velocity template. **Note:** This field appears only if the **Type** is selected as **Others**.

e. Click **Next Step**.

The **Endpoint Query Parameters** screen is displayed.

Figure 4-7 Endpoint Query Parameters



f. Specify the fields on **Endpoint Query Parameters** screen.

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

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

Table 4-6 Endpoint Query Parameters - Field Description

Field	Description
Name	Specify the name of the header. **Note:** This field appears only if the **Type** is selected as **Others**.
Value	Specify the value of the header. Value can be hardcoded or velocity template. **Note:** This field appears only if the **Type** is selected as **Others**.

g. Click **Add** for adding it in service list.

Parameter Group
Parameter mapping is used to establish the relationship between parameters of 2 different systems i.e., consumer and provider.

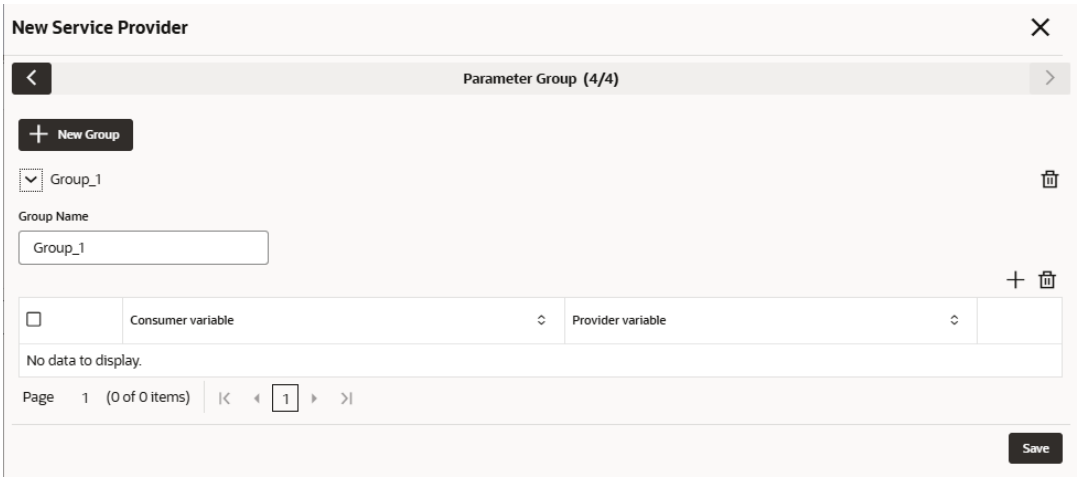
So, you can use consumer's parameter to find the corresponding parameter of provider and vice versa.

8. Click **Next**.

   The **New Service Provider - Parameter Group** screen is displayed.

Figure 4-8 New Service Provider - Parameter Group

'



For fetching provider parameter using consumer parameter,
Syntax: $custom.getParameterValueByConsumerKey (groupName, consumerParameter)

For fetching consumer parameter using provider parameter,
Syntax: $custom.getParameterValueByProviderKey (groupName, providerParameter)

Import Service Provider

9. Click **Import**.

   The **Import Service Provider** screen is displayed.

Figure 4-9 Import Service Provider

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

Table 4-7 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.

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

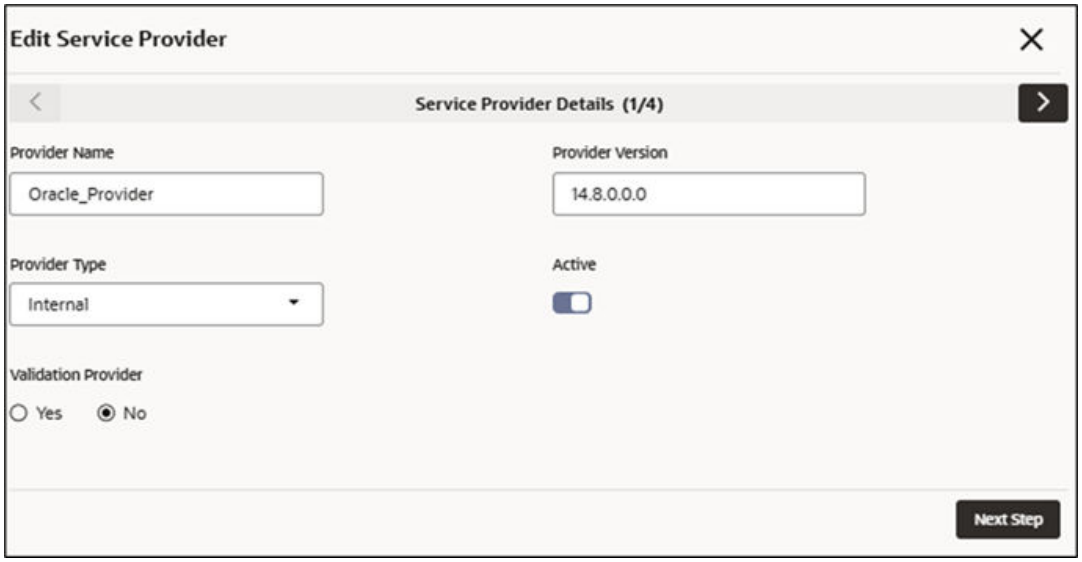
> **Note:**
>
> The following data needs to be changed after importing provider configuration file:
>
> - Implementation Host and Port
> - Implementation Authentication Password

View / Edit Service Provider

11. On **Service Provider** tile, click **View More** , and click **Edit Service Provider**.

    The **Edit Service Provider - Service Provider Details** screen is displayed.

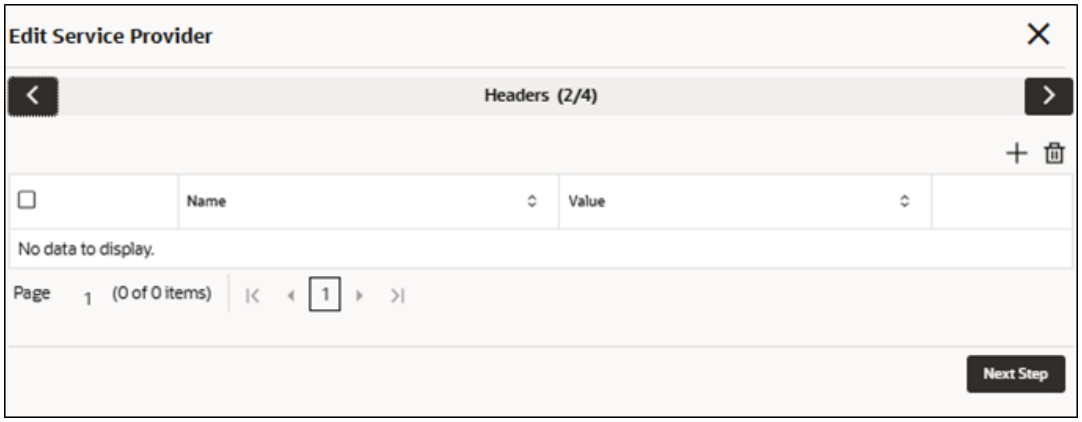
Figure 4-10 Edit Service Provider - Service Provider Details



12. Click **Next Step**.

    The **Edit Service Provider - Headers** screen is displayed.

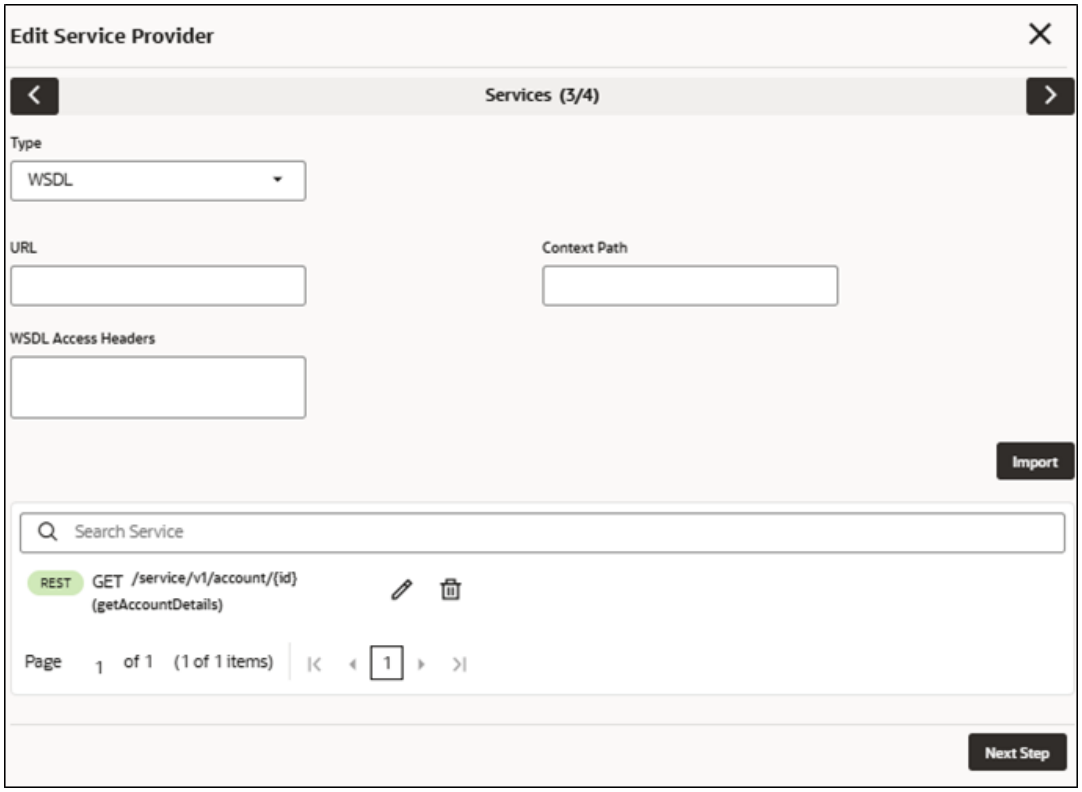
Figure 4-11 Edit Service Provider - Headers



13. Click **Next Step**.

    The **Edit Service Provider - Services** screen is displayed.

Figure 4-12 Edit Service Provider - Services



14. Click **Next Step**.

    The **Edit Service Provider - Parameter Group** screen is displayed.

Figure 4-13 Edit Service Provider - Parameter Group

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

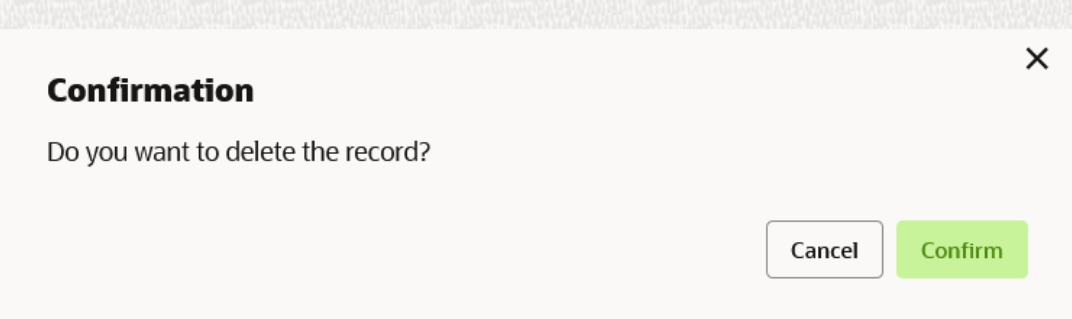
Delete Service Provider

The user can delete the provider.

16. On **Service Provider** tile, click **Delete** icon.

    The **Confirmation** screen is displayed.

Figure 4-14 Confirmation - Delete



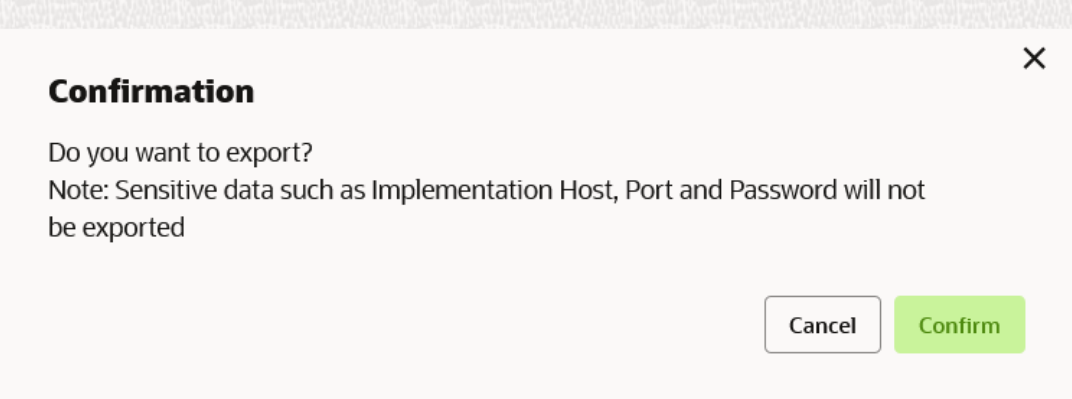
17. Click **Confirm** to delete the selected Service Provider.

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

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

    The **Confirmation** screen is displayed.

Figure 4-15 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.

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

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

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

    The **Configuration** screen is displayed.

Figure 4-16 Configuration

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

Table 4-8 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.

Table 4-8 (Cont.) Configuration Service Provider - Field Description

Field	Description
Read Timeout	This property is used to set the timeout on waiting to read data. **Note:** Value should be in milliseconds.
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. Value should be in milliseconds.
Keep-Alive duration	This property is used to keep connection alive for that specific time in connection pool before closing it. Value should be in milliseconds.

Request Audit

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

    The **Request Audit** screen is displayed.

> **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.

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

5

Implementation

This topic provides the systematic instructions to configure the implementation.

The implementation includes an instance of the Eureka client, along with the host, port, authentication, and specific service details. The Oracle Banking Routing Hub is compatible with web services and REST APIs.

> **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 is displayed.

Figure 5-1 Implementation

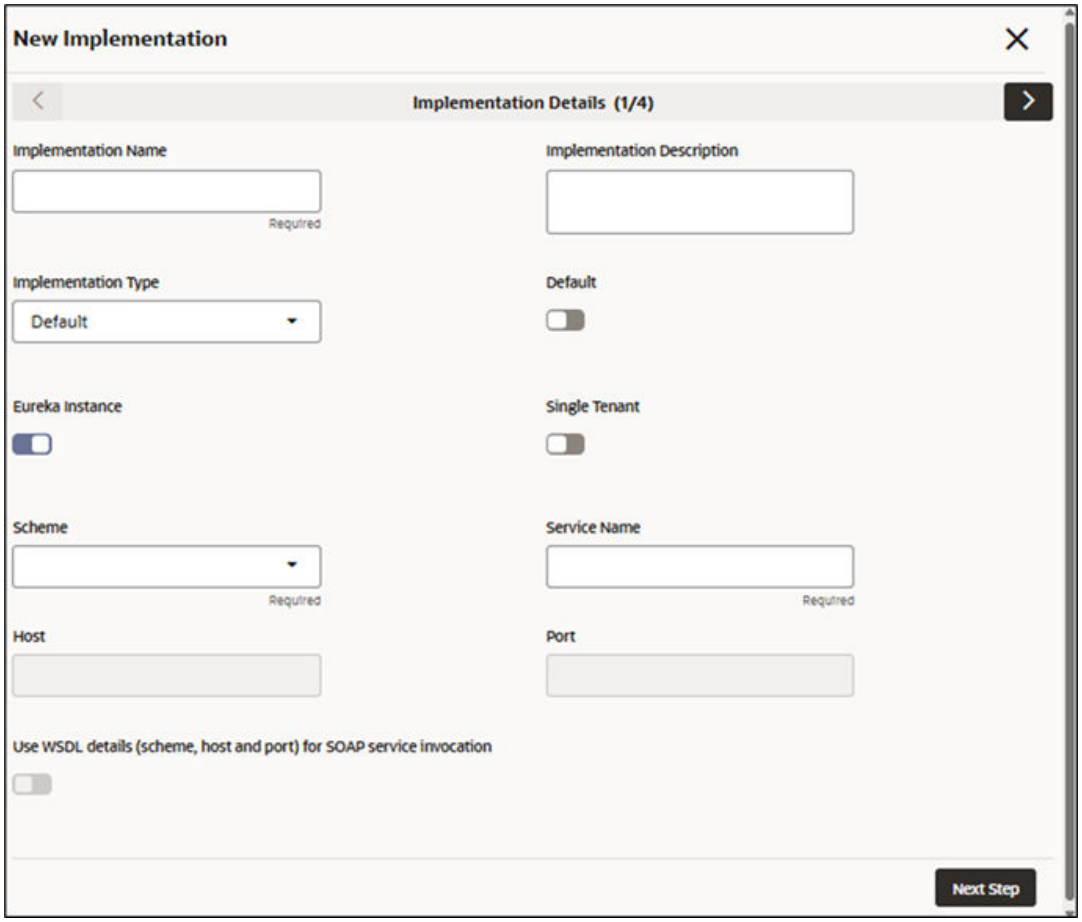
Implementation Details

The user can create the implementation manually.

2. Click **New**.

   The **New Implementation - Implementation Details** screen is displayed.

Figure 5-2 New Implementation - Implementation Details



3. On **New Implementation - Implementation Details** screen, specify the fields.

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

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

Table 5-1 New Implementation - Implementation Details - Field Description

Field	Description
Implementation 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 5-1 (Cont.) New Implementation - Implementation Details - Field Description

Field	Description
Implementation Description	Specify the description of the implementation. **Note:** • Enter 0 to 1000 characters. • No space allowed at beginning or ending of the characters.
Implementation Type	Select the type of implementation from drop-down list. The available options are: • **DEFAULT** • **QUEUE** • **OIC** • **SOCKET** • **KAFKA** **DEFAULT** type is for REST and SOAP API calls. **Note:** The type as **OIC** is only applicable for cloud services.
Default	Toggle the button if user wants to default. Each type can have one default implementation.
Single Tenant	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.

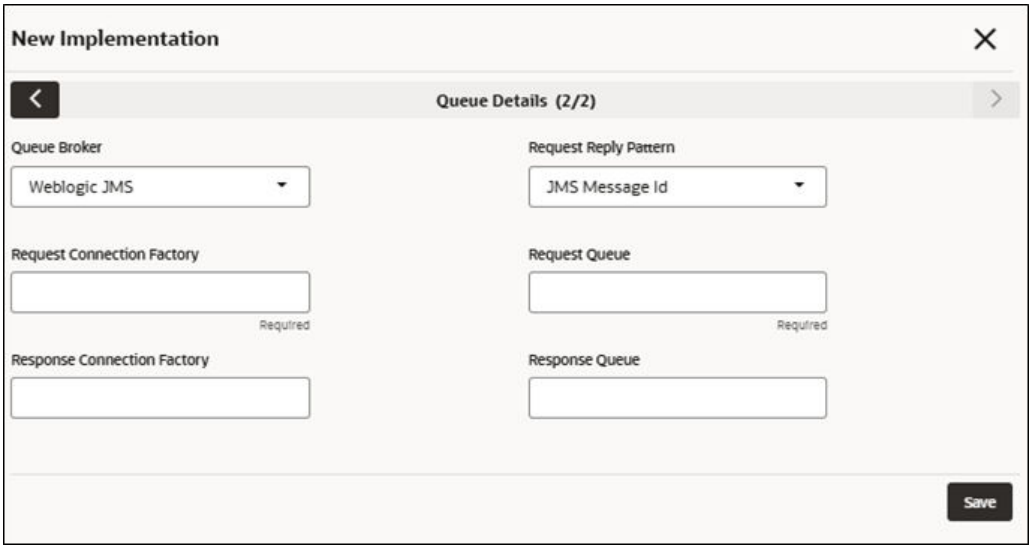
Table 5-1 (Cont.) New Implementation - Implementation Details - Field Description

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.
Use WSDL details (scheme, host and port) for SOAP service	This property is for using WSDL's scheme, host and port details for SOAP service invocation. Instead of using SOAP address's scheme, host and port details appearing in WSDL.

a. On **New Implementation** screen, for adding queue details manually, select the **Implementation Type** as **Queue** to define the queue details.

The **Queue Details** screen is displayed.

Figure 5-3 Queue Details



b. Specify the fields on **Queue Details** screen.

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

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

Table 5-2 Queue Details - Field Description

Field	Description
Queue Broker	Select the queue broker from drop-down list. The available options are:**WEBLOGIC_JMS**
Request Reply Pattern	Select the queue broker from drop-down list. The available options are: • **JMS_MESSAGEID** • **JMS_CORRELATIONID** **JMS_MESSAGEID**is default request-reply pattern.
Request Connection Factory	Specify the connection factory. Connection Factory is JNDI based connection factory name which is used to create connection for JMS client.
Request Queue	Specify the queue. Queue Name is JNDI based destination name.
Response Connection Factory	Specify the connection factory. Response Connection Factory is needed when destination is going to respond back after processing the request.

Table 5-2 (Cont.) Queue Details - Field Description

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

c. On **New Implementation** screen, for adding queue details manually, select the **Implementation Type** as **Kafka** to define the queue details.

The **Kafka Details** screen is displayed.

Figure 5-4 Kafka Details

d. Specify the fields on **Kafka Details** screen.

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

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

Table 5-3 Kafka Details - Field Description

Field	Description
Topic Name	Specify the topic name for publishing the message.

e. On **New Implementation** screen, for adding queue details manually, select the **Implementation Type** as **OIC** to define the queue details.

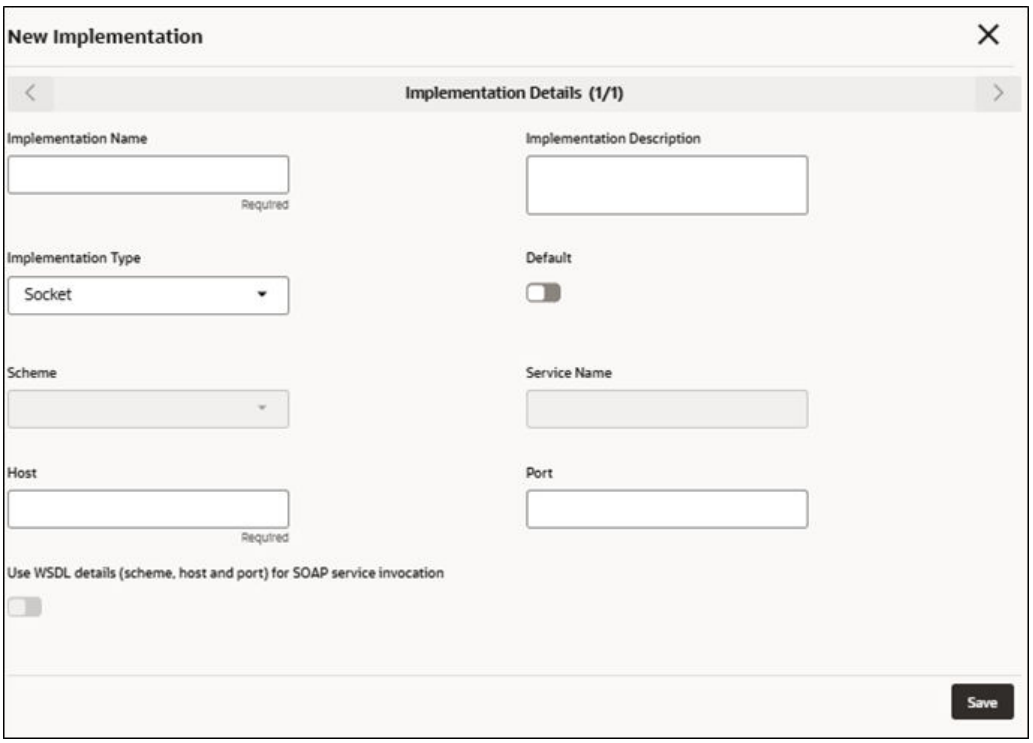
The **OIC** screen is displayed.

Figure 5-5 OIC

f. On **New Implementation** screen, for adding queue details manually, select the **Implementation Type** as **Socket** to define the queue details.

The **Socket** screen is displayed.

Figure 5-6 Socket



Authentication Details

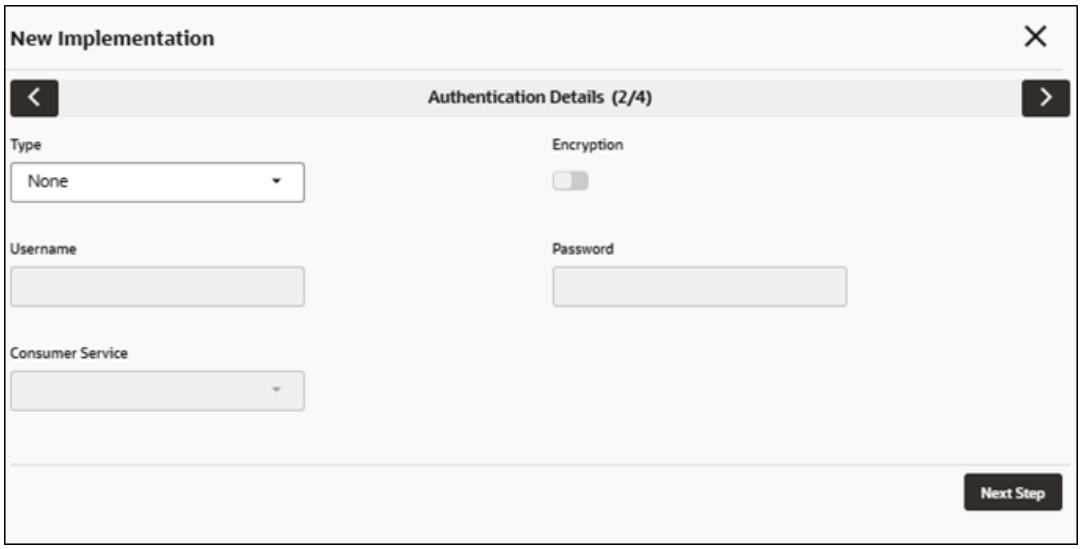
If external product processor require authentication to connect to it, Oracle Banking Routing Hub provides standard authentication mechanism schemes like Basic, JWT Token, OAuth Token, SSO, Custom.

> **Note:**
>
> If there is no authentication, set the Authentication Type to NONE. For identity propagation, set the Authentication Type to SSO. The token is cached for JWT Token, OAUTH_Token authentication type, and OIC Implementation Type.

4. Click **Next Step**.

   The **New Implementation - Authentication Details** screen is displayed.

Figure 5-7 New Implementation - Authentication Details



5. On **New Implementation - Authentication Details** screen, specify the fields.

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

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

Table 5-4 New Implementation - Implementation Details - Field Description

Field	Description
Type	Select the type of authentication from drop-down list. The available options are: • **Basic** • **JWT Token** • **OAUTH Token** • **SSO** • **Custom**

Table 5-4 (Cont.) New Implementation - Implementation Details - Field Description

Field	Description
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.
Consumer Service	Select the service which will be treated as custom authentication service. **Note:** **Custom Authentication** flag enabled consumer services are displayed

Headers

A provider implementation might require some standard headers to be passed along with the request. The user can specify the headers which are required by service endpoints but not present in swagger file.

Header step appears only if the Implementation **Type** is selected as **Default** or **OIC**.

Note:

Content-type header will be removed from Provider request if header value is NONE.

6. Click **Next Step**.

The **New Implementation - Headers** screen is displayed.

Figure 5-8 New Implementation - Headers

7. Specify the fields on the **New Implementation - Headers** screen.

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

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

Table 5-5 New Implementation - Headers - Field Description

Field	Description
Name	Specify the name of the header.
Value	Specify the value of the header. Value can be hardcoded or velocity template.

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.

  Context path can be modified for existing WSDL operations.

  > **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 and OpenAPI 3.0 both are supported.

Existing REST endpoints can also be modified or deleted.

> **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.

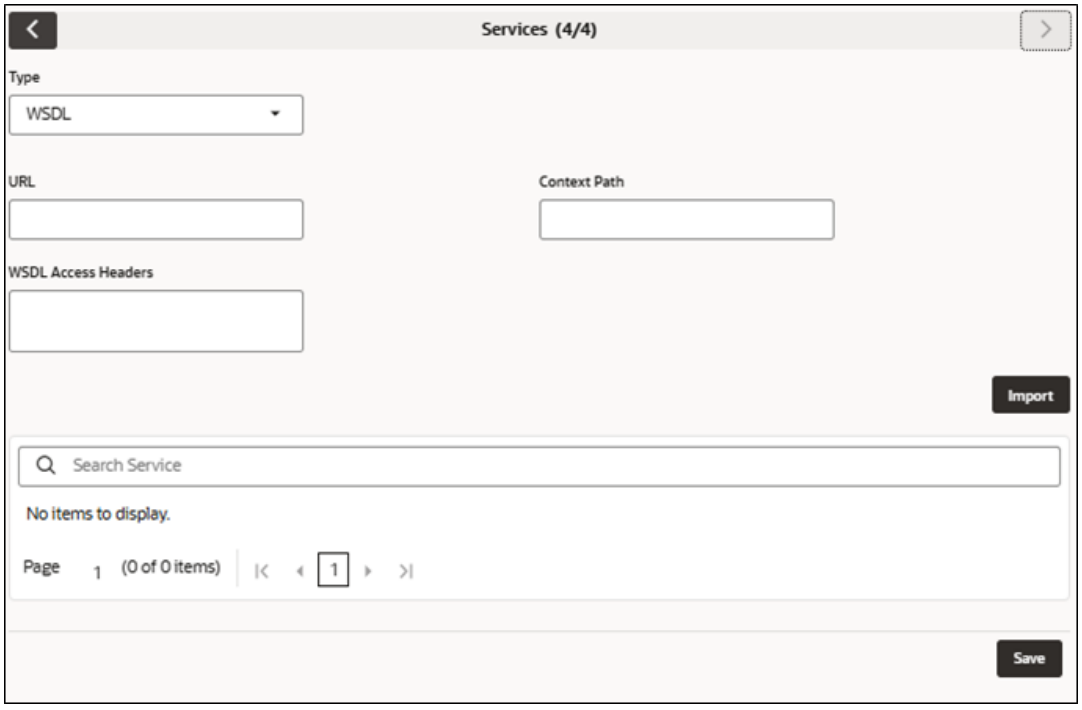
- **OTHERS**:
  OTHERS option is selected for adding REST API details manually when provider does not have swagger file.

> **Note:**
>
> If there is a change in existing endpoint, then the same endpoint details can be modified using edit option.

8. Click **Next Step**.

   The **New Implementation - Services** screen is displayed.

Figure 5-9 New Implementation - Services



9. Specify the fields on the **New Implementation - Services** screen.

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

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

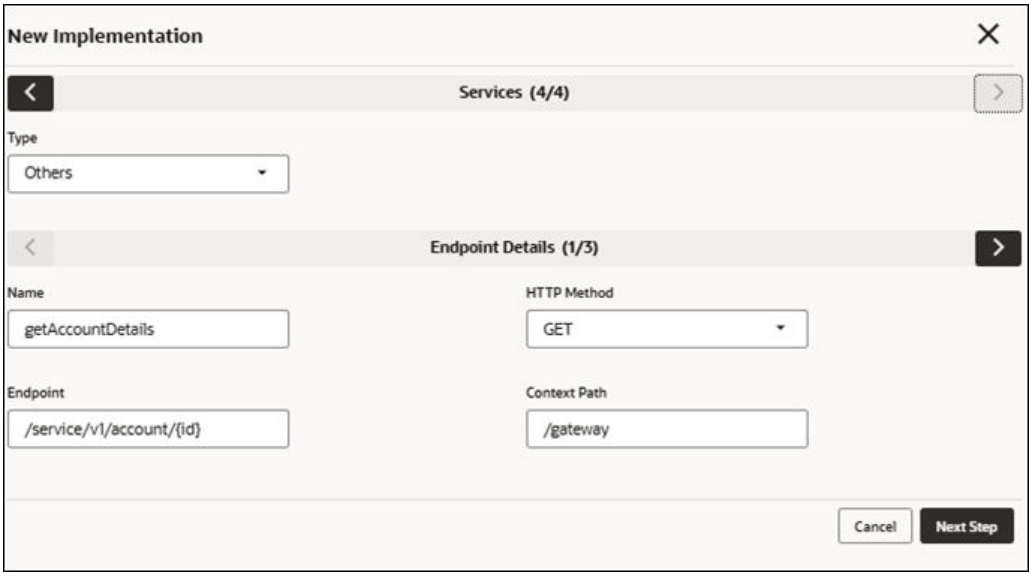
Table 5-6 New Implementation - Services - Field Description

Field	Description
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**
URL	Specify the service URL of the file location. **Note:** This field appears only if the **Type** is selected as **WSDL** and **SWAGGER**.
Content path Prefix	Context path of below formatted URL. http://host:port/context-path/endpointGateway
WSDL Access Headers	Specify the headers required for accessing / reading WSDL's.
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**.

a. On **New Implementation** screen, for adding REST endpoint details manually, select the **Type** as **Others** to define the endpoint details.

   The **Endpoint Details** screen is displayed.

Figure 5-10 Endpoint Details



b. Specify the fields on **Endpoint Details** screen.

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

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

Table 5-7 Endpoint Details - Field Description

Field	Description
Name	Specify the name of the operation. **Note:** This field appears only if the **Type** is selected as **Others**.

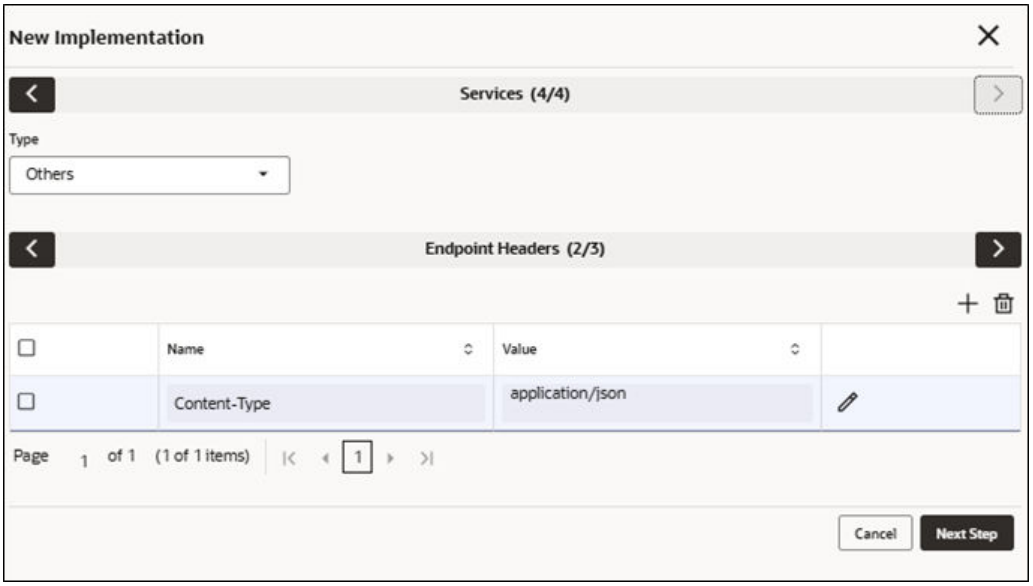
Table 5-7 (Cont.) Endpoint Details - Field Description

Field	Description
HTTP Method	Select the HTTP method from the drop-down list. The available options are: • **GET** • **POST** • **PUT** • **PATCH** • **DELETE** **Note:** This field appears only if the **Type** is selected as **Others**.
Endpoint URL	Specify the endpoint URL for the operation. **Note:** This field appears only if the **Type** is selected as **Others**.
Content Path	Context path of below formatted URL http://host:port/context-path/endpoint

c. Click **Next Step**.

The **Endpoint Headers** screen is displayed.

Figure 5-11 Endpoint Headers



d. Specify the fields on **Endpoint Headers** screen.

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

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

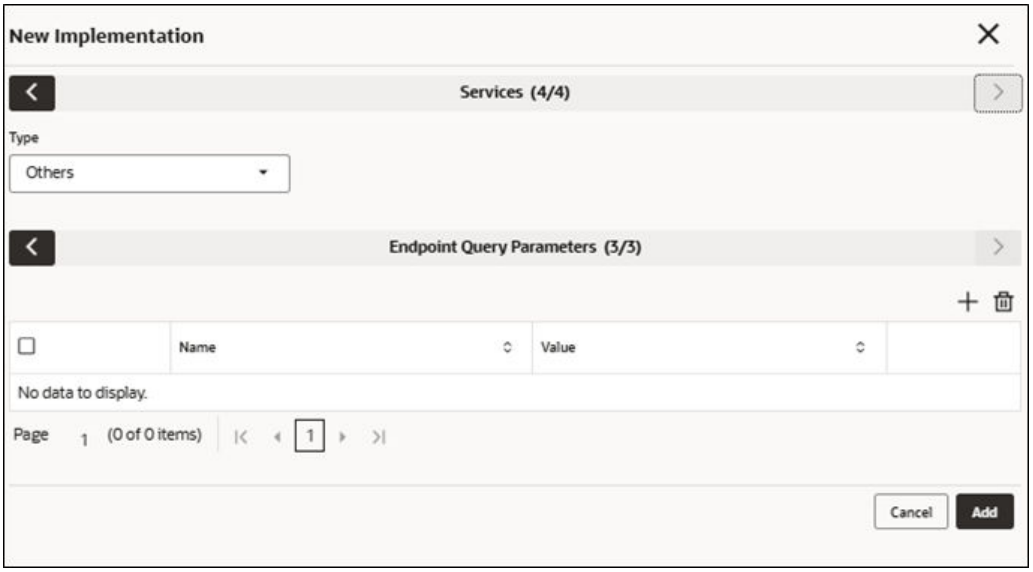
Table 5-8 Endpoint Headers - Field Description

Field	Description
Name	Specify the name of the header. **Note:** This field appears only if the **Type** is selected as **Others**.
Value	Specify the value of the header. Value can be hardcoded or velocity template. **Note:** This field appears only if the **Type** is selected as **Others**.

e. Click **Next Step**.

The **Endpoint Query Parameters** screen is displayed.

Figure 5-12 Endpoint Query Parameters



f. Specify the fields on **Endpoint Query Parameters** screen.

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

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

Table 5-9 Endpoint Query Parameters - Field Description

Field	Description
Name	Specify the name of the header. **Note:** This field appears only if the **Type** is selected as **Others**.
Value	Specify the value of the header. Value can be hardcoded or velocity template. **Note:** This field appears only if the **Type** is selected as **Others**.

g. Click **Add** for adding it in service list.

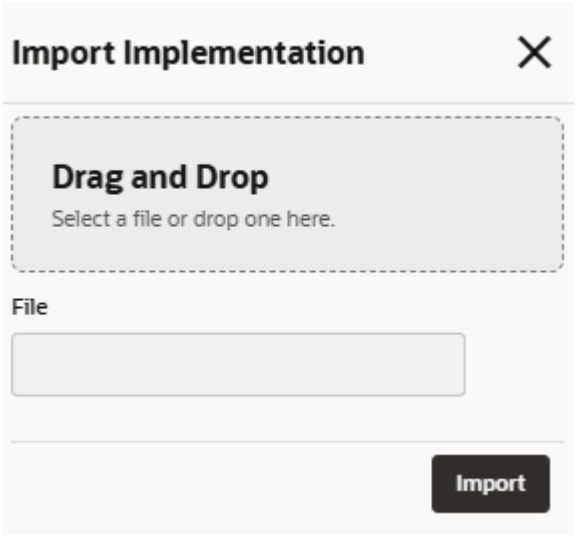
Import Implementation

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).

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

    The **Import Implementation** screen is displayed.

Figure 5-13 Import Implementation



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

Table 5-10 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.

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

    The below data needs to be changed after importing implementation configuration file:

    - Implementation Host and Port
    - Implementation Authentication Password

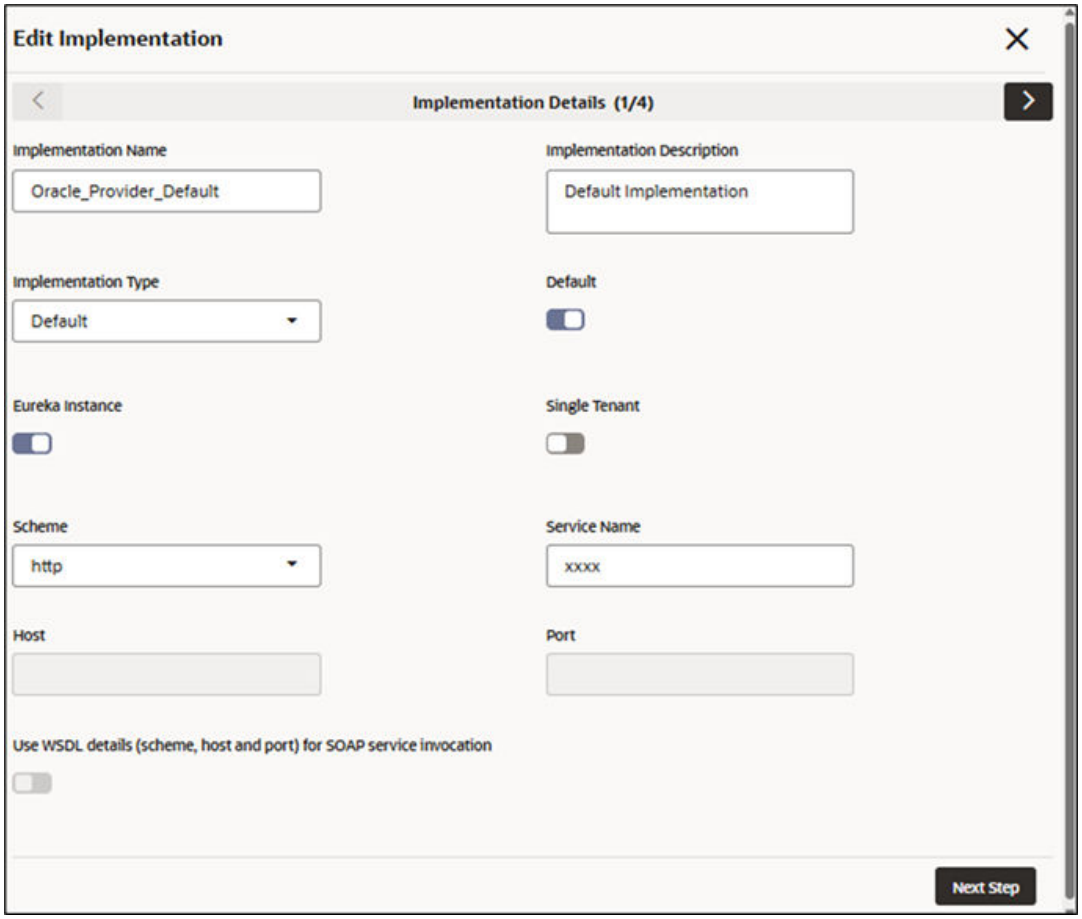
View / Edit Implementation

The user can view or modify implementation details.

12. On **Implementation** screen, click **Edit** icon .

    The **Edit Implementation** screen is displayed.

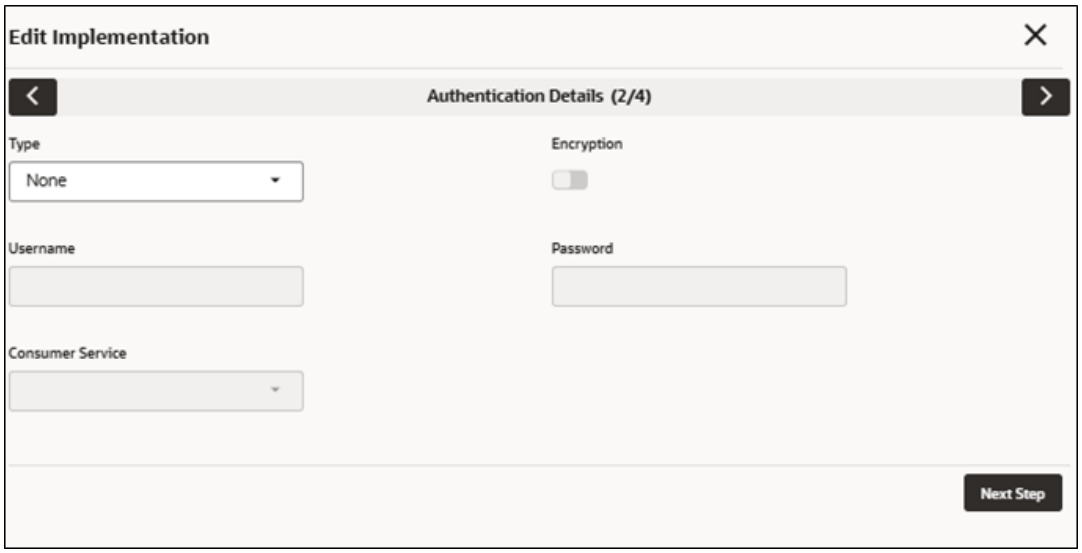
Figure 5-14 Edit Implementation - Implementation Details



13. Click **Next Step**.

    The **Edit Implementation - Authentication Details** screen is displayed.

Figure 5-15 Edit Implementation - Authentication Details



14. Click **Next Step**.

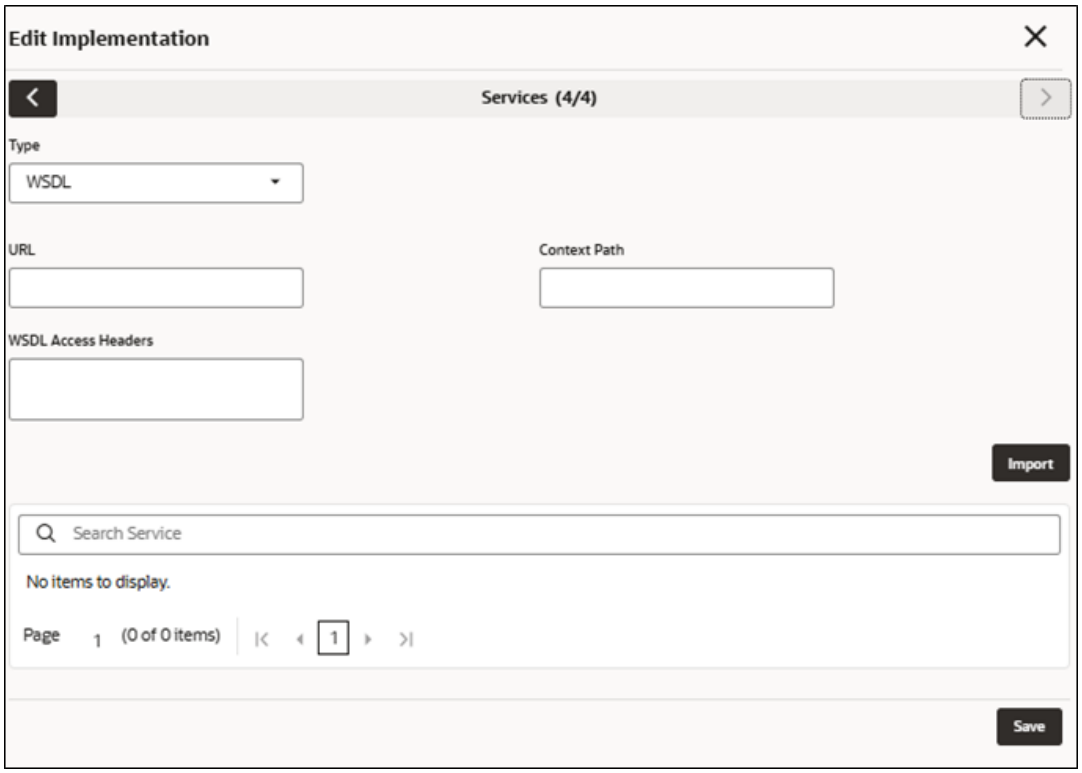
    The **Edit Implementation - Headers** screen is displayed.

Figure 5-16 Edit Implementation - Headers

15. Click **Next Step**.

    The **Edit Implementation - Services** screen is displayed.

Figure 5-17 Edit Implementation - Services



16. 7. Click **Save** to save the modified implementation details.

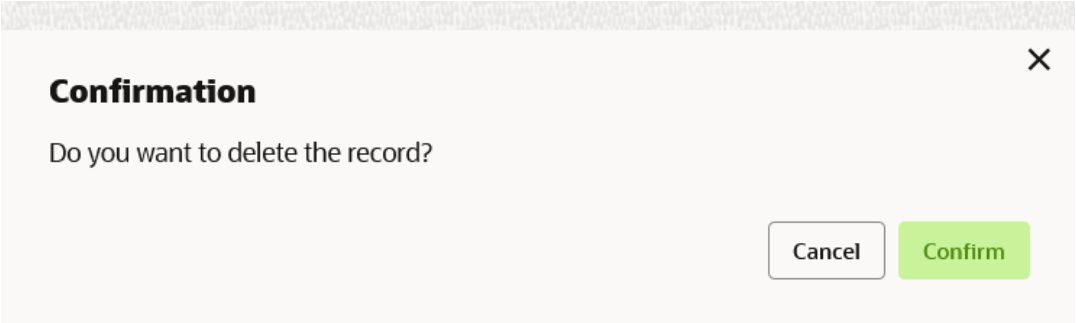
Delete Implementation

The user can delete the implementation details.

17. On **Implementation** screen, click **Delete**.

    The **Confirmation** screen is displayed.

Figure 5-18 Confirmation - Delete

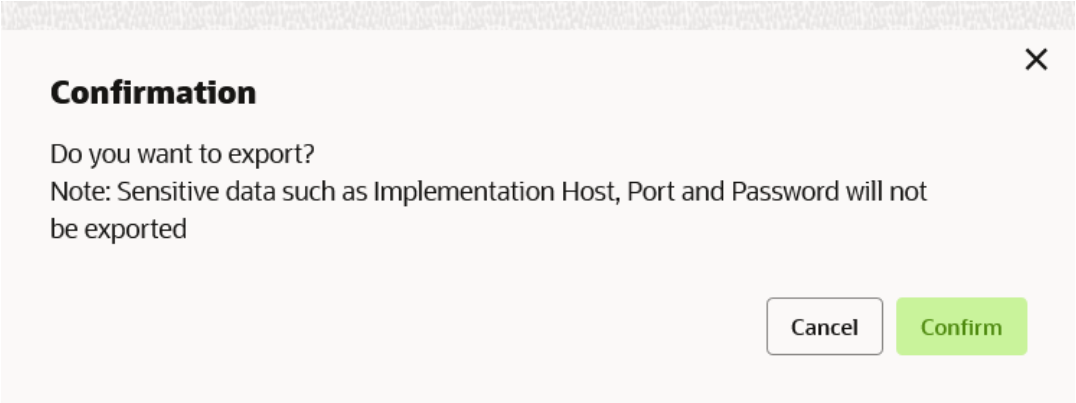


Export Implementation
The user can export the implementation configuration as JSON file.

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

    The **Confirmation** screen is displayed.

Figure 5-19 Confirmation - Export Implementation



Below data cannot be exported:

- Implementation Host
- Implementation Port
- Implementation Authentication Password

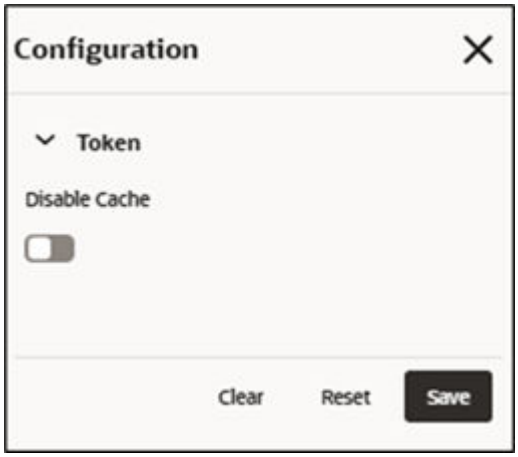
The above data needs to be configured manually after importing the configuration file. Same has been mentioned in Import section.

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

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

    The **Configuration** screen is displayed.

Figure 5-20 Configuration



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

Table 5-11 Configuration - Field Description

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

Request Audit

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

    The **Request Audit** screen is displayed.

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

Clear Cache

The user can clear the SOAP client cache.

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

6
Consumer Services

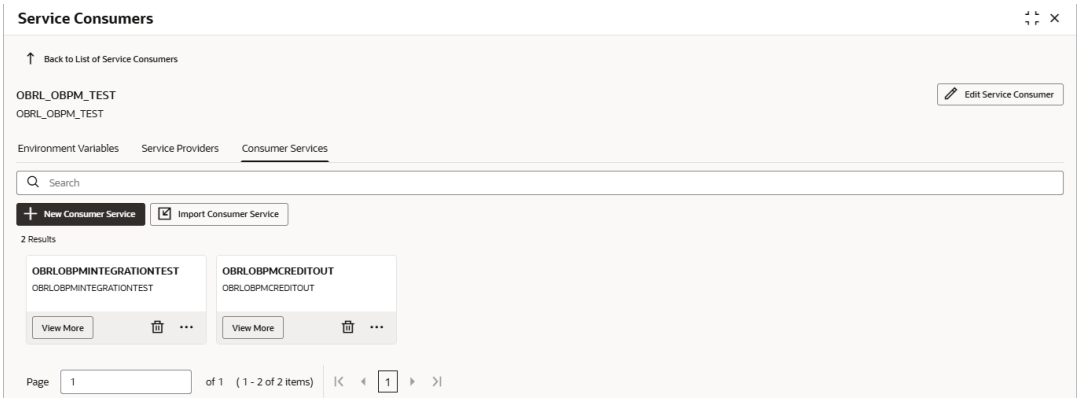
This topic describes the systematic instructions to configure the consumer services.

Consumer Services specifies the service ID that is transmitted by the service consumer. It also handles transitions and route definitions, including the details for source integration.

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

   The **Consumer Services** screen is displayed.

Figure 6-1 Consumer Services

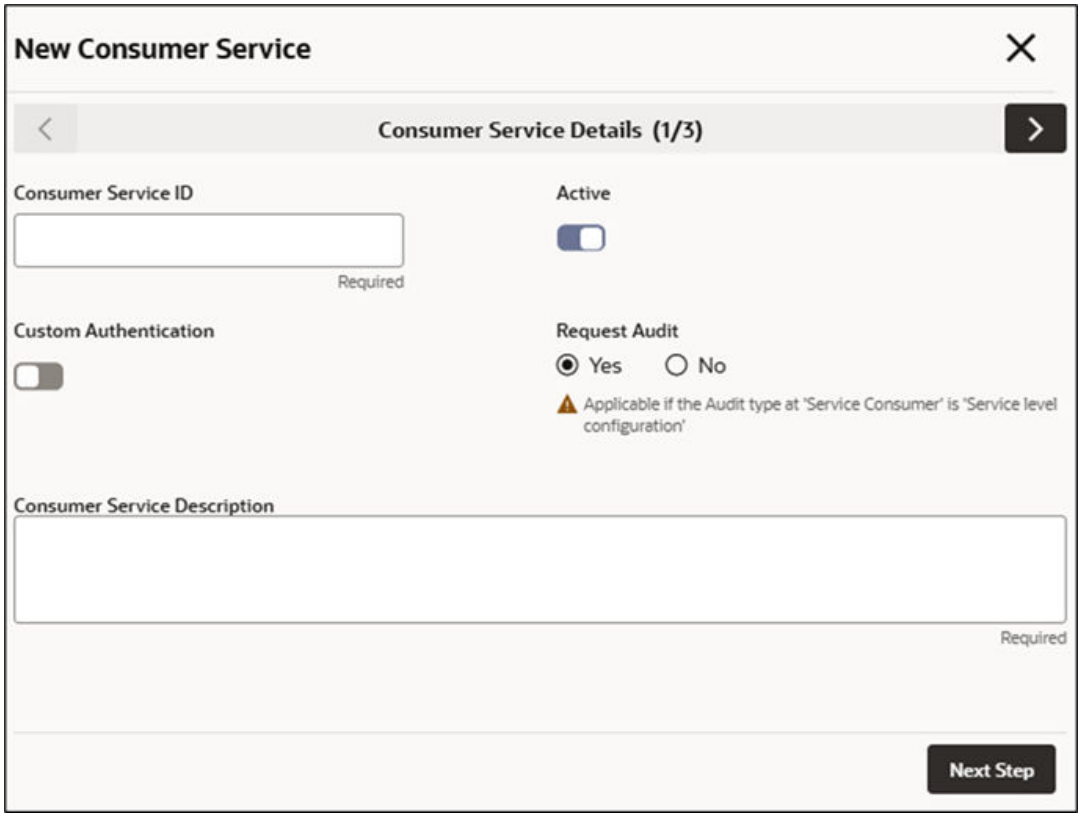


New Consumer Service
The user can create Consumer Service manually.

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

   The **New Consumer Service - Consumer Service Details** screen is displayed.

Figure 6-2 New Consumer Service - Consumer Service Details



3. On **New Consumer Service - Consumer Service Details** screen, specify the fields.

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

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

Table 6-1 New Consumer Service - Consumer Service Details - Field Description

Field	Description
Consumer Service 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.
Active	ON / OFF If this flag is toggled OFF, then all related routes will be stopped.
Custom Authentication	This flag is to mark the consumer service which can be used as custom authentication service in implementation.

Table 6-1 (Cont.) New Consumer Service - Consumer Service Details - Field Description

Field	Description
Request Audit Properties	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**'
Consumer Service Description	Specify the description of the consumer service. **Note:** • Enter 0 to maximum of 1000 characters. • No space allowed at beginning or ending of the characters.

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

      The **Attributes** screen is displayed.

Figure 6-3 Attributes

   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 6-2 Add Header - Field Description

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

Note:

- Using $.body, the user can access the request body.
  Syntax: $.body.fieldName

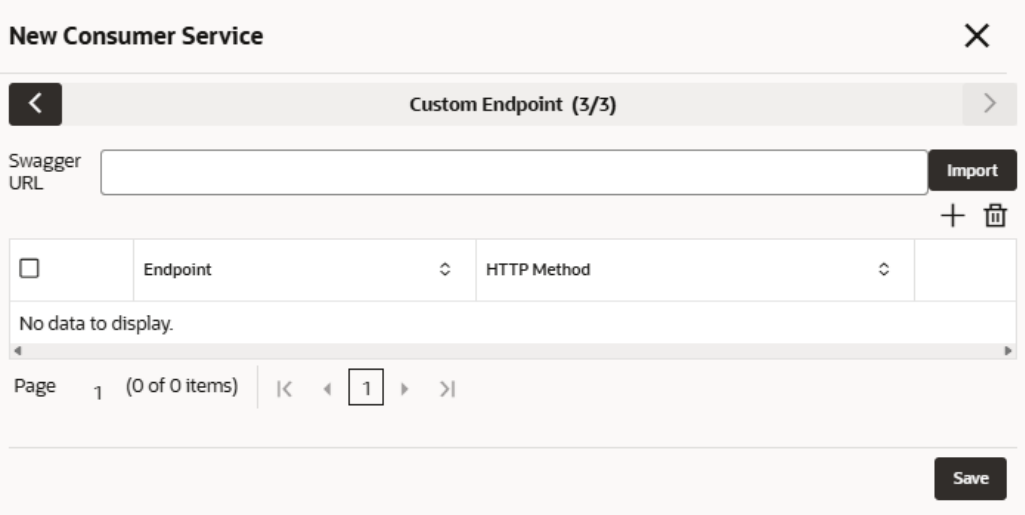
  Example: $.body.branchCode
- Using $.headers, the user can access the request headers.
  Syntax: $.headers["fieldname "][0]

  Example: $.headers["branchCode"][0]
- Using $.env, the user can access the environment variables.
  Syntax: $.env.group.variable

c. Click **Next Step**.

The **Custom Endpoint** screen is displayed.

Figure 6-4 Custom Endpoint



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

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).

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

   The **Import Service** screen is displayed.

Figure 6-5 Import Service

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

Table 6-3 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.

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

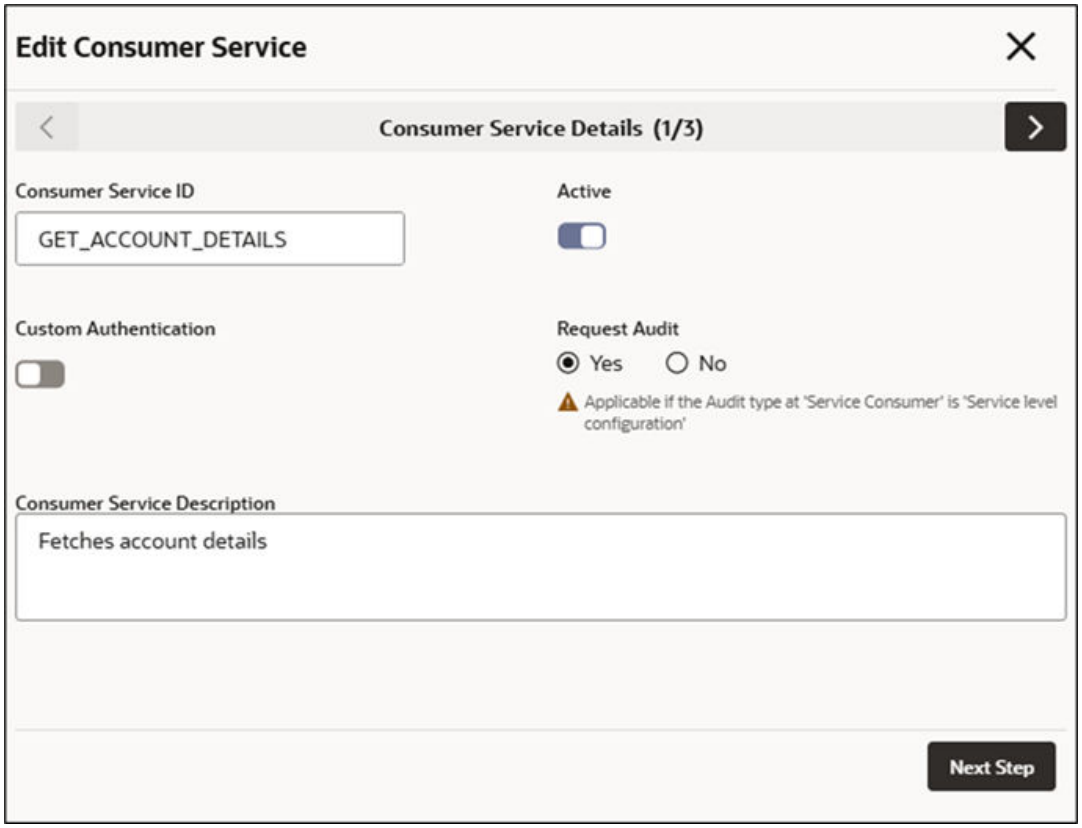
View / Edit Consumer Service

The user can view or modify consumer service details.

8. On **Consumer Service** tile, click **View More** , and click **Edit Consumer Service**.

   The **Edit Consumer Service** screen is displayed.

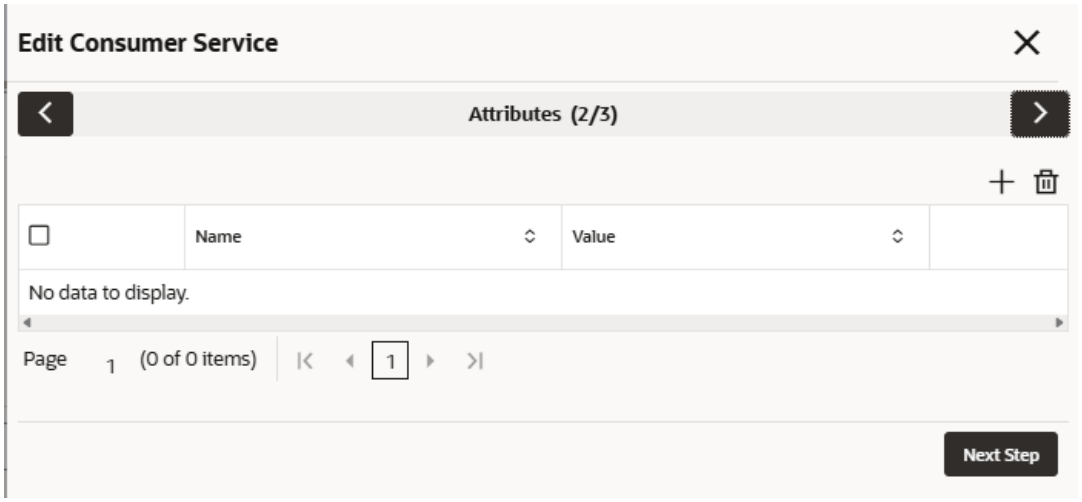
Figure 6-6 Edit Consumer Service



9. Click **Next Step**

The **Edit Consumer Service - Attributes** screen is displayed.

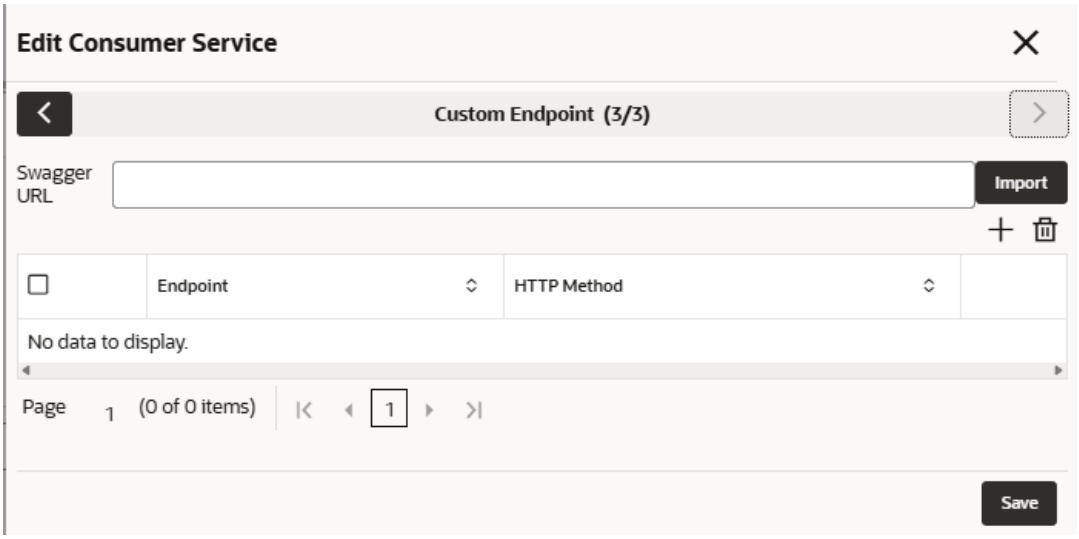
Figure 6-7 Edit Consumer Service - Attributes



10. Click **Next Step**

The **Edit Consumer Service - Custom Endpoint** screen is displayed.

Figure 6-8 Edit Consumer Service - Custom Endpoint



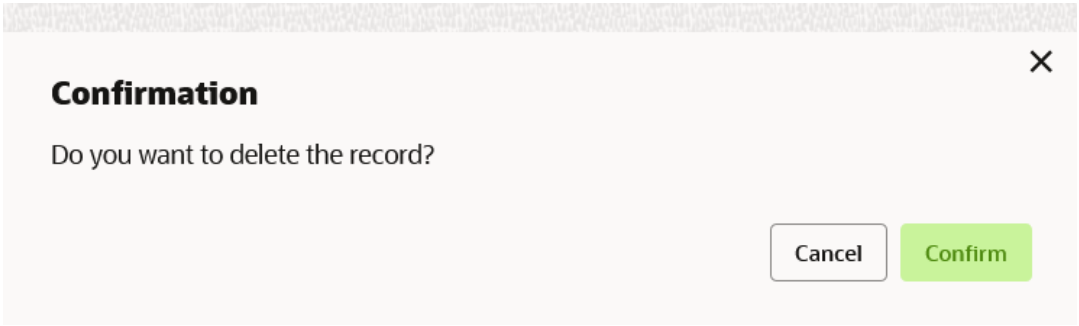
11. Click **Save** save the modified consumer service details.

Delete Consumer Service
The user can delete the consumer service.

12. On **Consumer Service** tile, click **Delete**.

    The **Confirmation** screen is displayed.

Figure 6-9 Confirmation

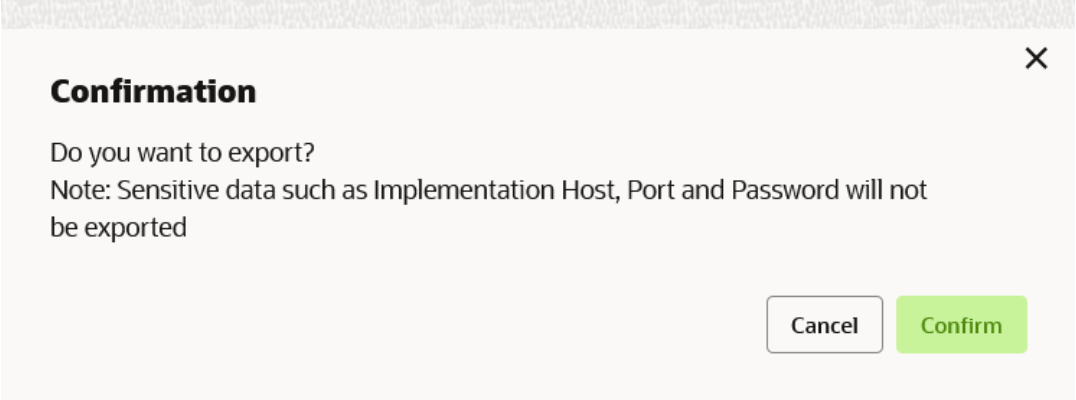


Export Consumer Service
The user can export the consumer service configuration as JSON file.

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

    The **Confirmation** screen is displayed.

Figure 6-10 Confirmation - Export



Consumer Service - Configuration

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

    The **Configuration** screen is displayed.

Figure 6-11 Consumer Service - Configuration

15. Specify the fields on **Consumer Service - Configuration** screen.

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

Table 6-4 Consumer Service - Configuration - Field Description

Field	Description
Service level timeout	This property is used to override the global and provider 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.

Note:

Refer to Configuration topic for the screen and field description of Route Shutdown properties.

Consumer Service - Request Audit

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

    The **Request Audit** screen is displayed.

Note:

Refer to Request Audit topic for the screen and field description.

7
Transformation

This topic describes the systematic instructions to configure the transformation.

Transformation involves gathering and changing data from one source to another and back again. This process occurs within consumer services. It changes the data from the service consumer into a format suitable for the service provider.

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

   The **Transformation** screen is displayed.

Figure 7-1 Transformation

New Transformation
The user can create transformation manually.

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

   The **New Transformation - Basic Details** screen is displayed.

Figure 7-2 New Transformation - Basic Details

3. Specify the fields on **New Transformation - Basic Details** screen.

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

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

Table 7-1 New Transformation - Basic Details - Field Description

Field	Description
Transformation 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	ON / OFF If transformation is turned OFF, the user will be unable to choose transformation in routing.
Product Processor	Select the product processor from the drop-down list.
Implementation	Select the implementation from the drop-down list.

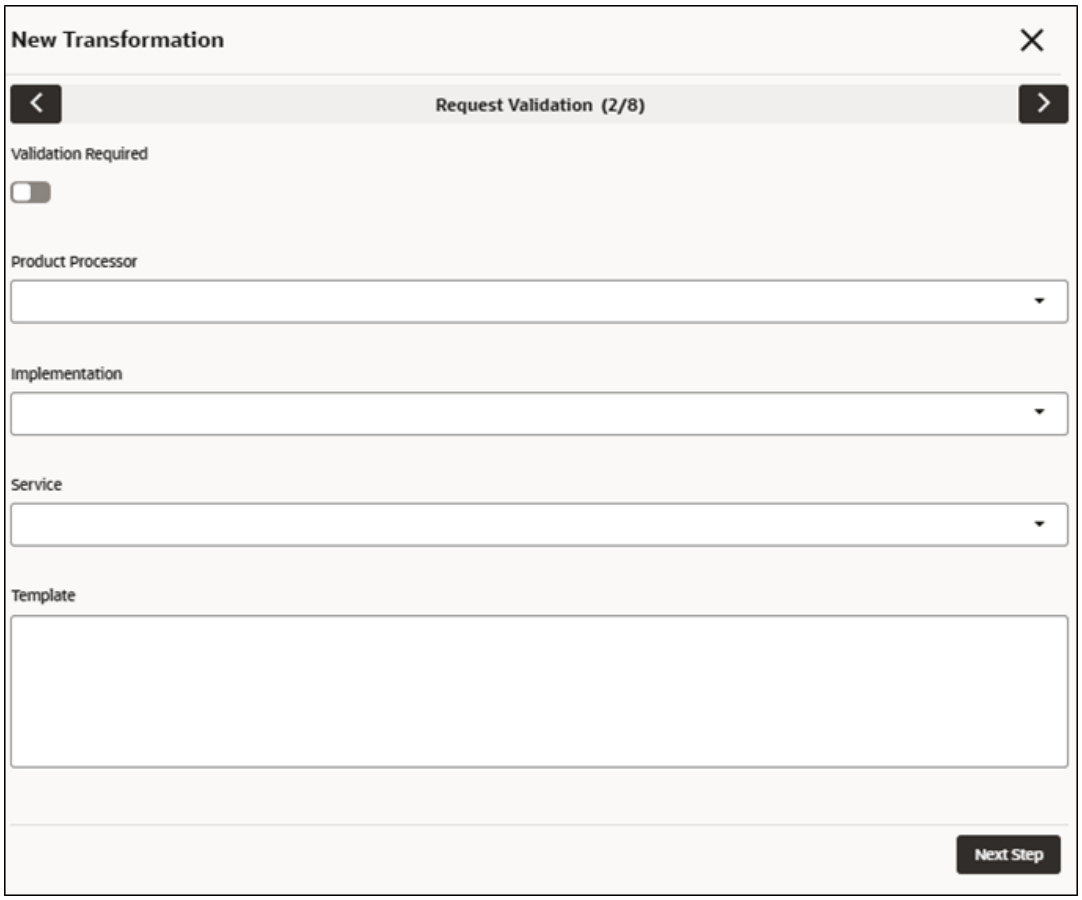
Table 7-1 (Cont.) New Transformation - Basic Details - Field Description

Field	Description
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.

4. Click **Next Step**.

   The**New Transformation - Request Validation** screen is displayed.

Figure 7-3 New Transformation - Request Validation



5. Specify the fields on **New Transformation - Request Validation** screen.

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

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

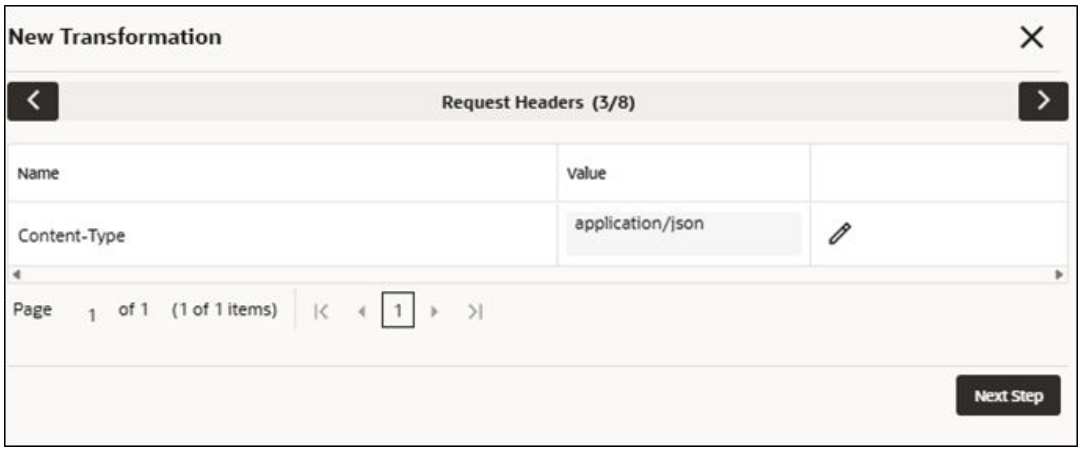
Table 7-2 New Transformation - Request Validation - Field Description

Field	Description
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.

6. Click **Next Step**.

   The**New Transformation - Request Headers** screen is displayed.

Figure 7-4 New Transformation - Request Headers



7. Specify the fields on **New Transformation - Request Headers** screen.

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

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

Table 7-3 New Transformation - Request Headers - Field Description

Field	Description
Name	A list of headers related to the chosen provider, implementation, and service is displayed. The user can only modify the header value.

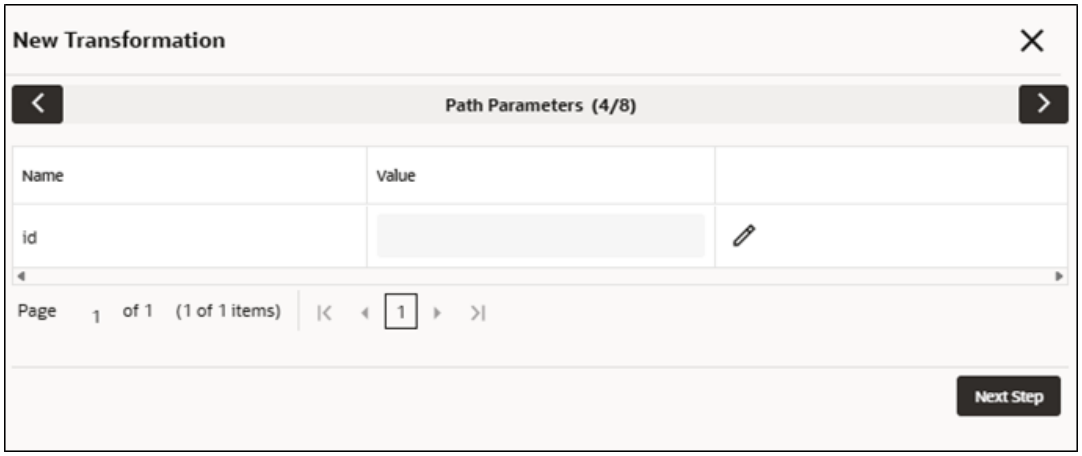
Table 7-3 (Cont.) New Transformation - Request Headers - Field Description

Field	Description
Value	Displays the value of the headers. Value can be hardcoded value or velocity mapping.

8. Click **Next Step**.

   The**New Transformation - Path Parameters** screen is displayed.

Figure 7-5 New Transformation - Path Parameters



9. Specify the fields on **New Transformation - Path Parameters** screen.

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

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

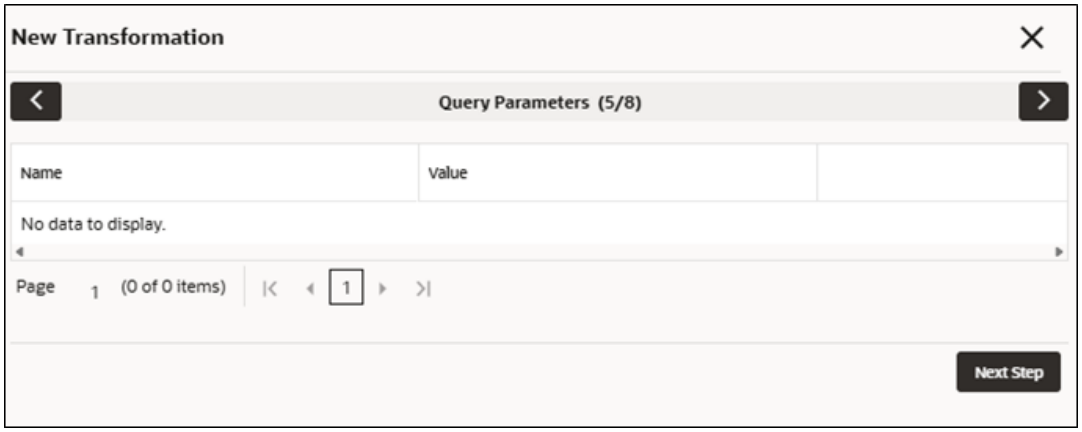
Table 7-4 New Transformation - Path Parameters - Field Description

Field	Description
Name	A list of path parameters related to the chosen service is displayed. User can only change the path parameter value.
Value	Displays the value of the headers. Value can be hardcoded value or velocity mapping.

10. Click **Next Step**.

    The**New Transformation - Query Parameters** screen is displayed.

Figure 7-6 New Transformation - Query Parameters



11. Specify the fields on **New Transformation - Query Parameters** screen.

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

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

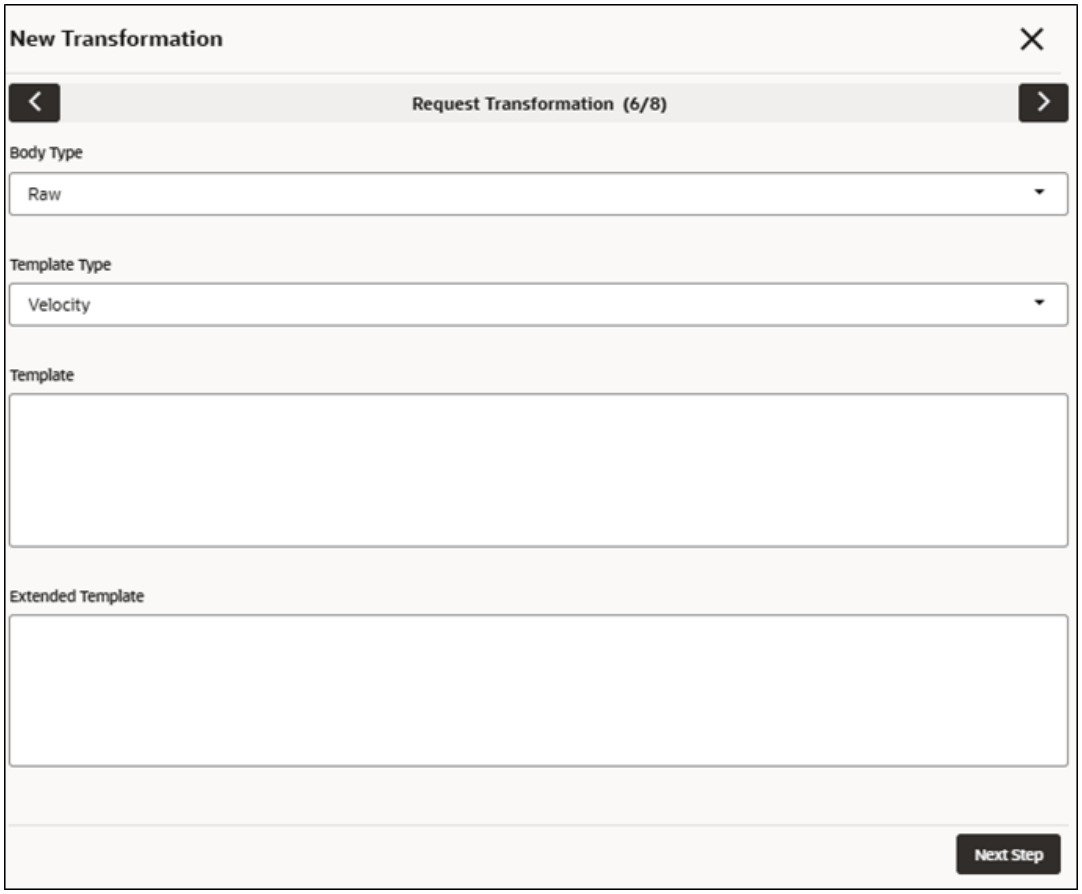
Table 7-5 New Transformation - Query Parameters - Field Description

Field	Description
Name	Query parameter list relevant to the selected service is displayed. User can only change the query parameter value.
Value	Displays the value of the headers. Value can be hardcoded value or velocity mapping.

12. Click **Next Step**.

The**New Transformation - Request Transformation** screen is displayed.

Figure 7-7 New Transformation - Request Transformation



13. Specify the fields on **New Transformation - Request Transformation** screen.

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

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

Table 7-6 New Transformation - Request Transformation - Field Description

Field	Description
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**.

14. Click **Next Step**.

    The**New Transformation - Response Headers** screen is displayed.

Figure 7-8 New Transformation - Response Headers

15. Specify the fields on **New Transformation - Response Headers** screen.

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

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

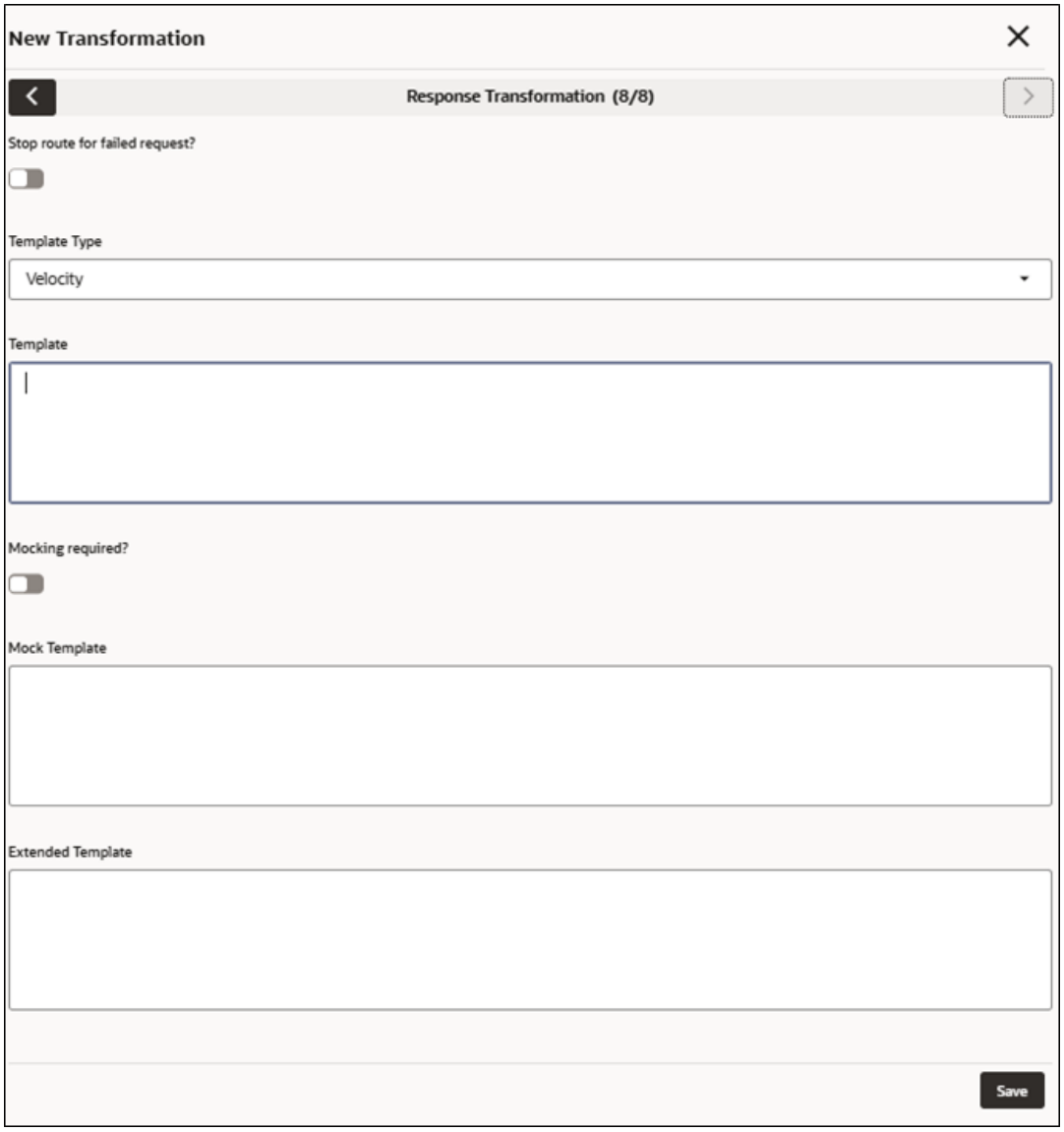
Table 7-7 New Transformation - Response Headers - Field Description

Field	Description
Name	User can specify the additional headers that are required to be part of Routing Hub response headers.
Value	Displays the value of the headers. Value can be hardcoded value or velocity mapping.

16. Click **Next Step**.

The**New Transformation - Response Transformation** screen is displayed.

Figure 7-9 New Transformation - Response Transformation



17. Specify the fields on **New Transformation - Response Transformation** screen.

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

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

Table 7-8 New Transformation - Response Transformation - Field Description

Field	Description
Stop route for failed request	This property is used to handle response for failed request. **Note:** Only applicable for API chaining scenario.
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.
Extended Template	Specify the custom template in order to extend the kernel template. Refer to Extensibility and Transformation Type for syntax.

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

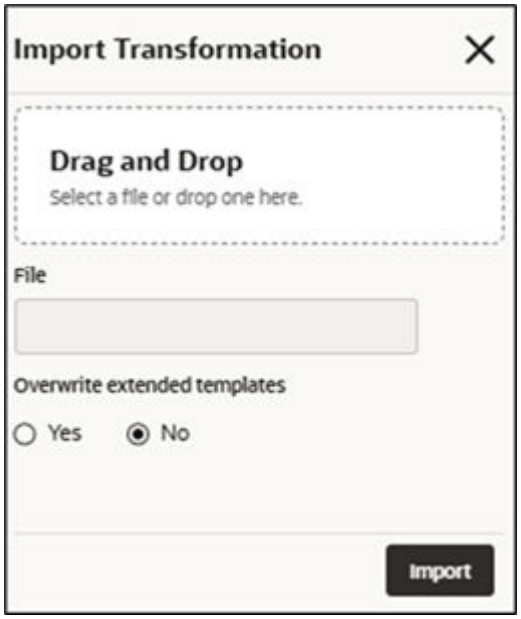
Import Transformation

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).

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

    The **Import Transformation** screen is displayed.

Figure 7-10 Import Transformation



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

Table 7-9 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.

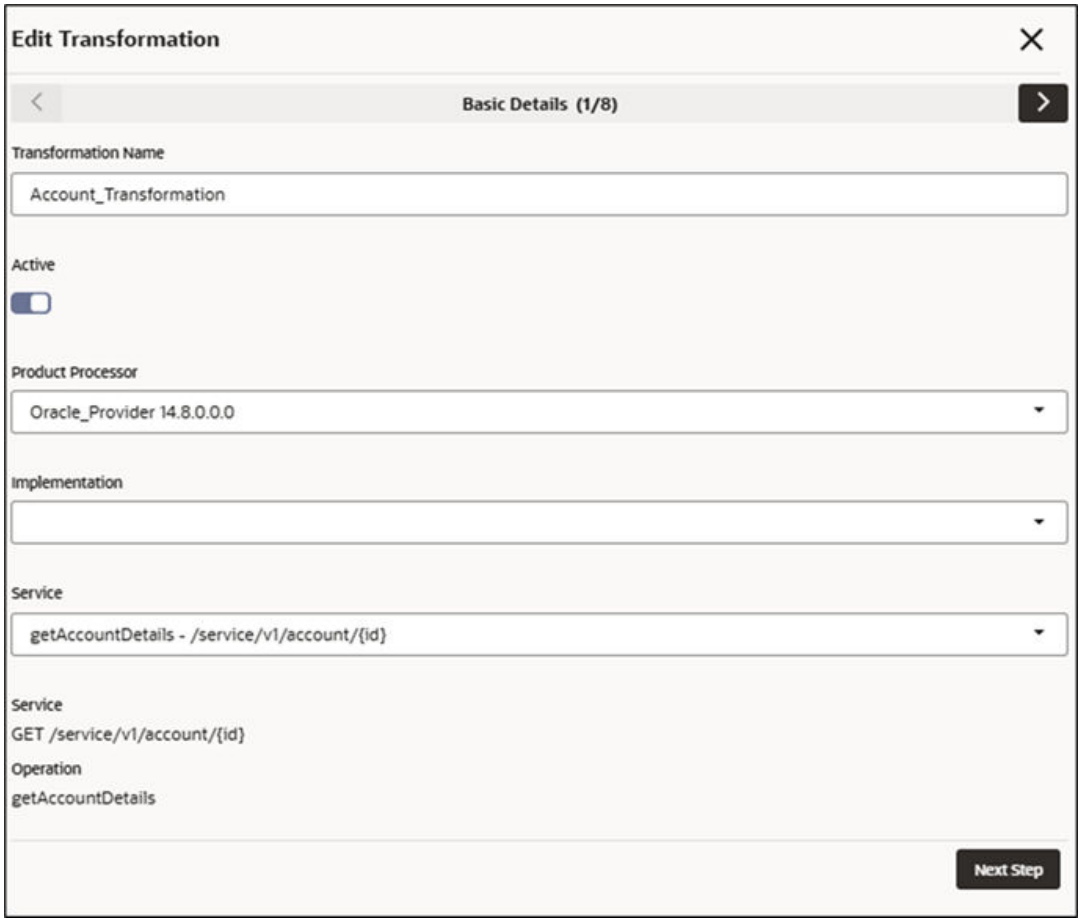
20. Click **Import** to import the selected file.

View / Edit Transformation
The user can view or more transformation details.

21. On **Transformation** list, click **Edit**.

    The **Edit Transformation - Basic Details** screen is displayed.

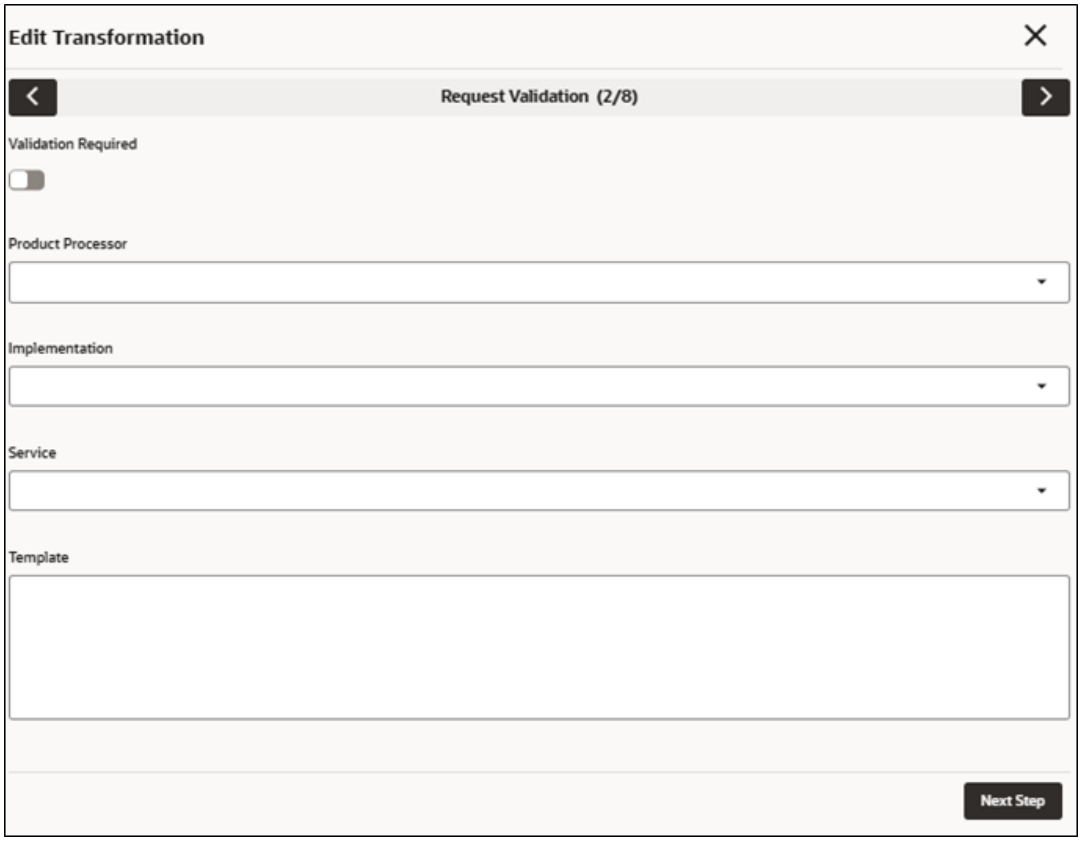
Figure 7-11 Edit Transformation - Basic Details



22. Click **Next Step**.

    The **Edit Transformation - Request Validation** screen is displayed.

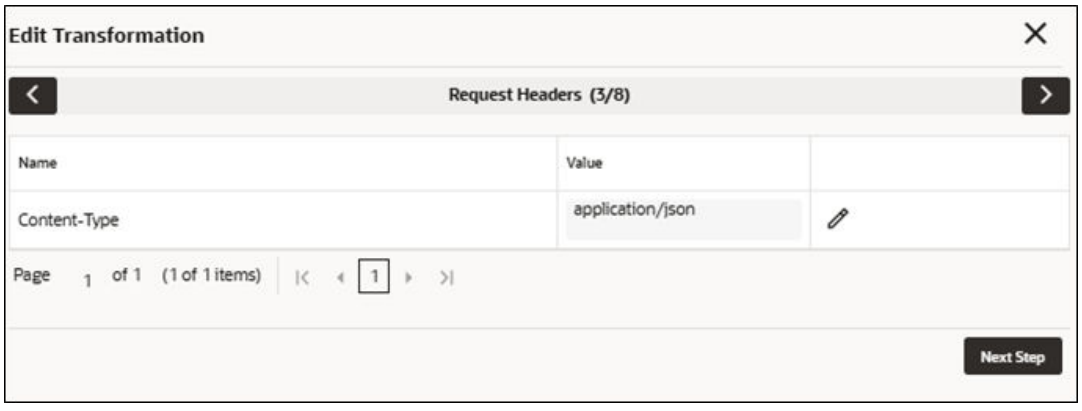
Figure 7-12 Edit Transformation - Request Validation



23. Click **Next Step**.

    The **Edit Transformation - Request Headers** screen is displayed.

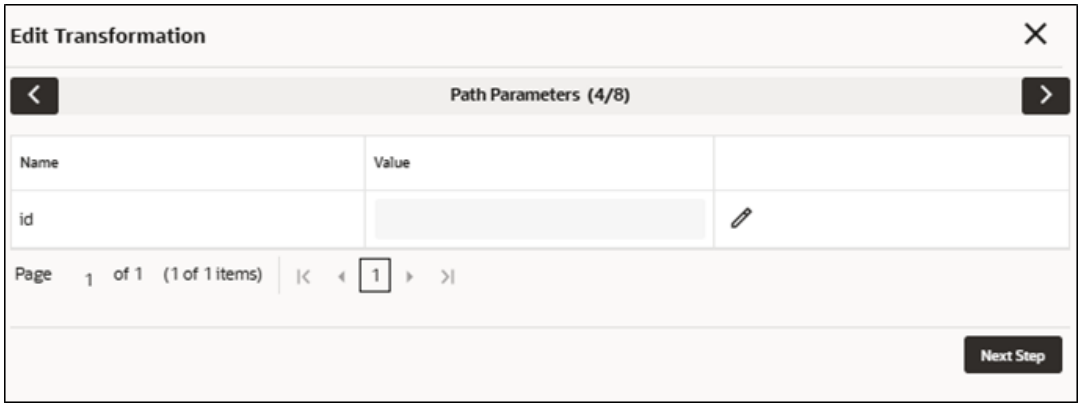
Figure 7-13 Edit Transformation - Request Headers



24. Click **Next Step**.

    The **Edit Transformation - Path Parameters** screen is displayed.

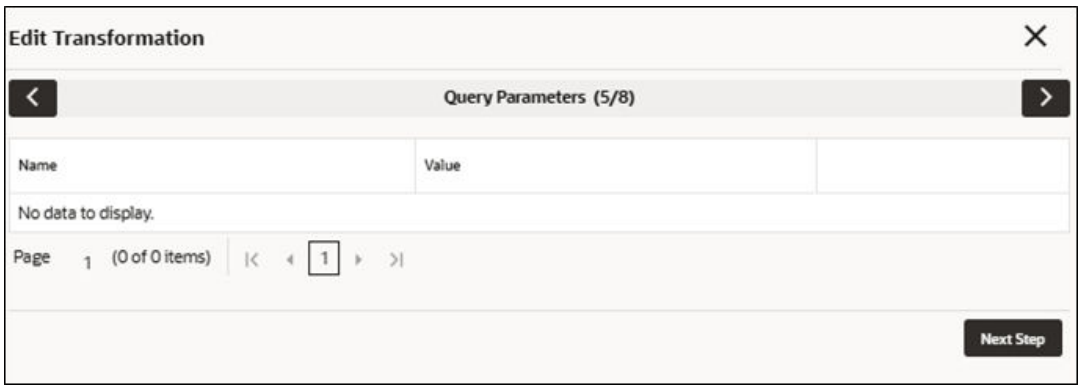
Figure 7-14 Edit Transformation - Path Parameters



25. Click **Next Step**.

    The **Edit Transformation - Query Parameters** screen is displayed.

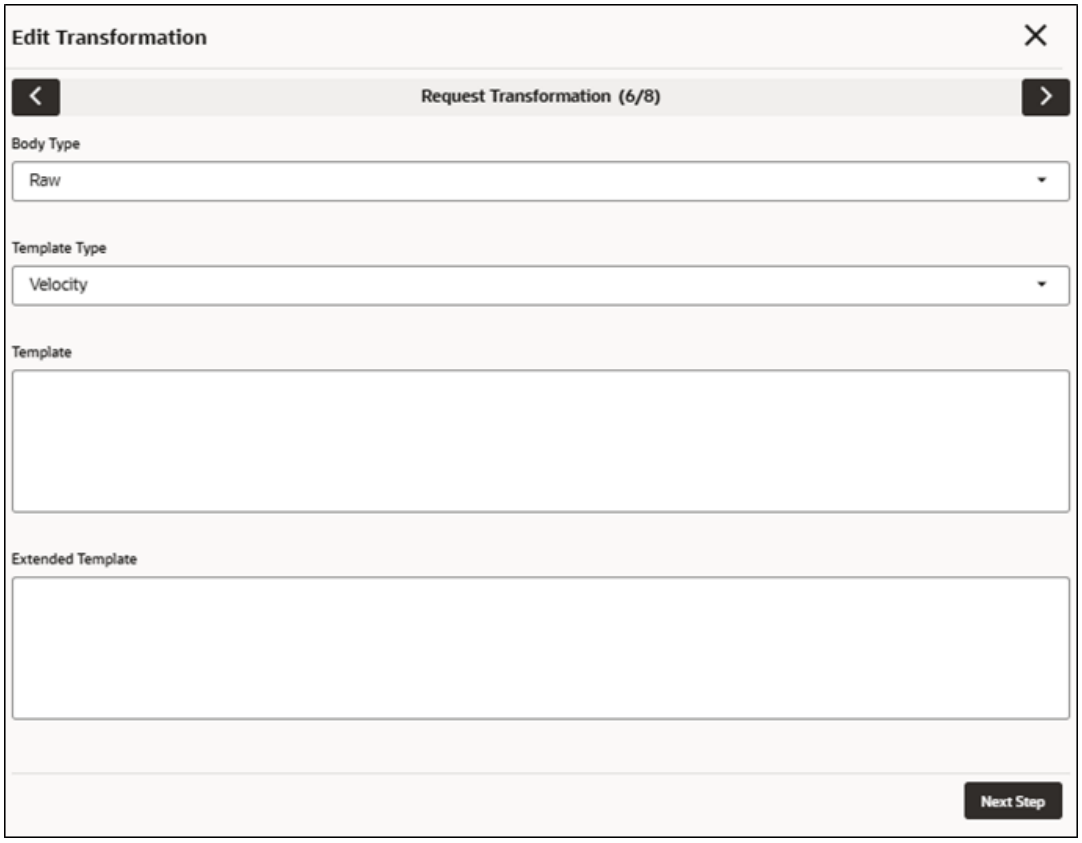
Figure 7-15 Edit Transformation - Query Parameters



26. Click **Next Step**.

    The **Edit Transformation - Request Transformation** screen is displayed.

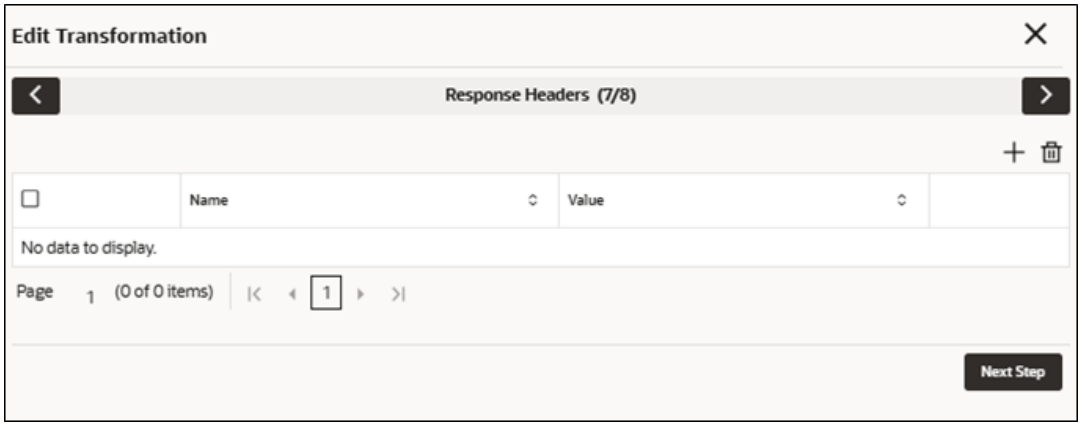
Figure 7-16 Edit Transformation - Request Transformation



27. Click **Next Step**.

    The **Edit Transformation - Response Headers** screen is displayed.

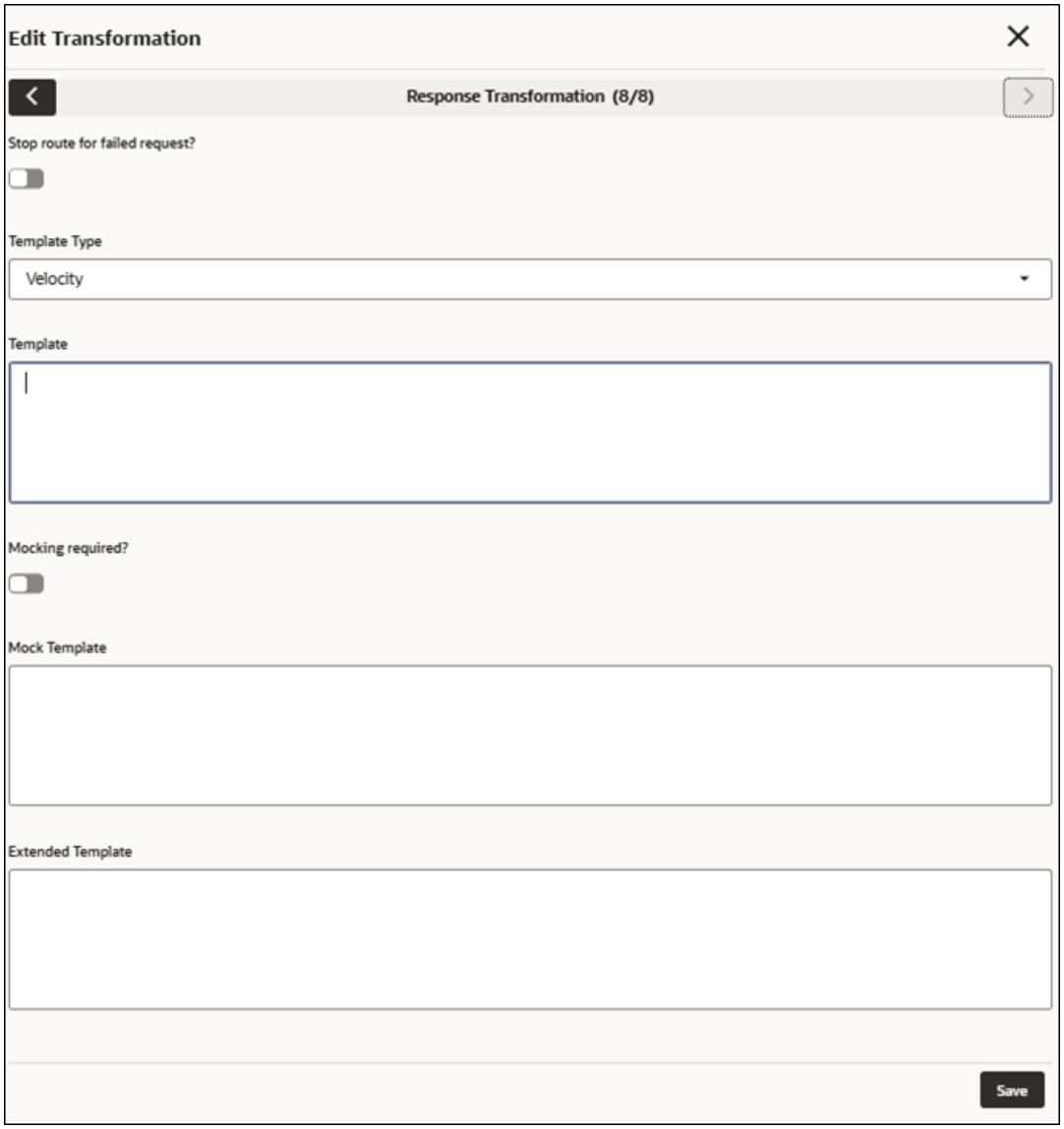
Figure 7-17 Edit Transformation - Response Headers



28. Click **Next Step**.

    The **Edit Transformation - Response Transformation** screen is displayed.

Figure 7-18 Edit Transformation - Response Transformation



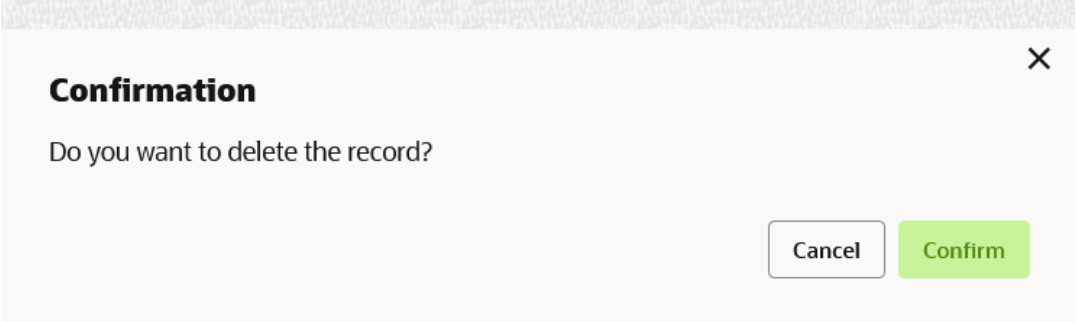
29. Click **Save** to save the modified transformation details.

Delete Transformation
The user can delete the transformation.

30. On **Transformation** list, click **Delete**.

    The **Confirmation - Delete** screen is displayed.

Figure 7-19 Confirmation - Delete



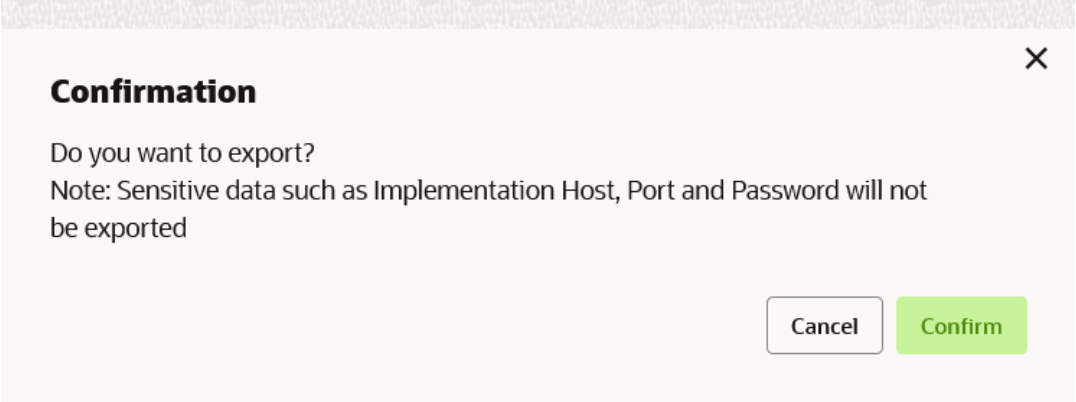
Export Transformation

The user can export the transformation configuration as JSON file.

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

    The **Confirmation** screen is displayed.

Figure 7-20 Confirmation - Export



Request Audit

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

    The **Request Audit** screen is displayed.

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

8
Routing

This topic describes the systematic instructions to configure the routing.

Routing does not establish any specific rules or configurations based on rules. Instead, it determines which service provider receives the actual request by considering maintenance and assessment factors.

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

   The **Routing** screen is displayed.

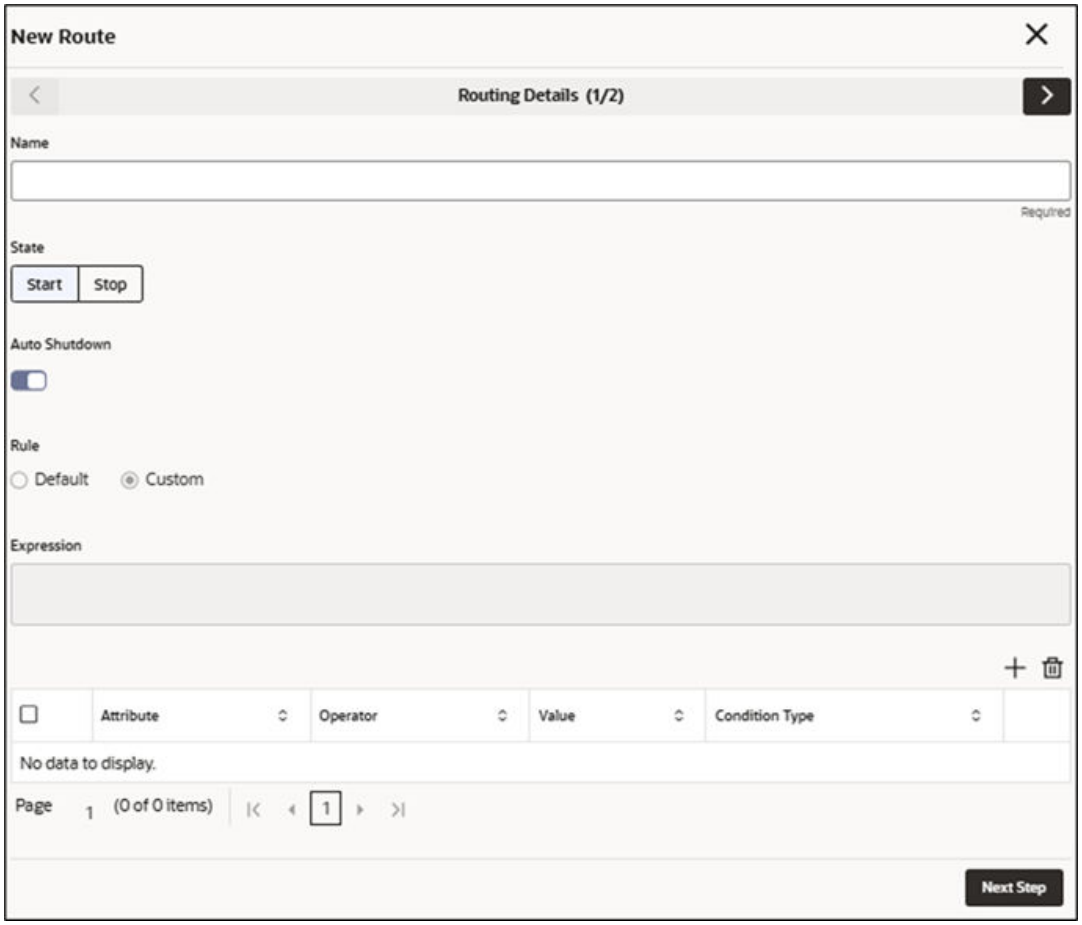
Figure 8-1 Routing

New Route
The user can create routing manually.

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

   The **New Route - Routing Details** screen is displayed.

Figure 8-2 New Route - Routing Details



3. On **New Route - Routing Details** screen, specify the fields.

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

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

Table 8-1 New Route - Routing Details - 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.

Table 8-1 (Cont.) New Route - Routing Details - Field Description

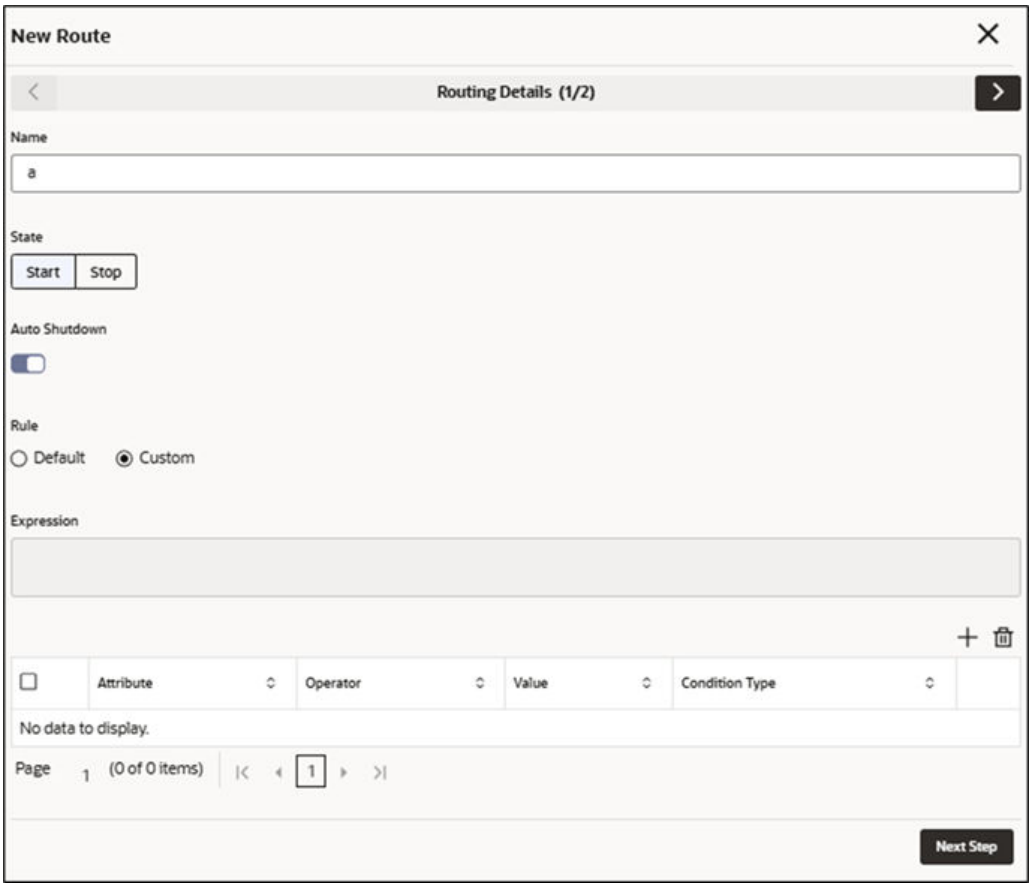
Field	Description
State	Start / Stop If routing is marked as STOP, then consumer request fails at routing hub level only.
Auto Shutdown	ON / OFF If the AutoShutdown flag is activated, the route state will switch to STOP if the route failure exceeds the allowed threshold limit set by the monitoring and 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 Custom Rule using Expression Attributes

4. To add rule, follow the below steps.
   a. On **New Route** screen, click **Custom** button.

      The **Expression Editor** screen is displayed.

Figure 8-3 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 8-2 Expression Editor - Field Description

Field	Description
Attribute	Select consumer service attribute 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.

> **Note:**
>
> String values must be enclosed in single quotes ('). For example: 'abc'. List values should be separated by commas and also enclosed in single quotes ('). For example: 'abc,xyz,1.23,true'.
> Environment variables can also be accessed using $env.

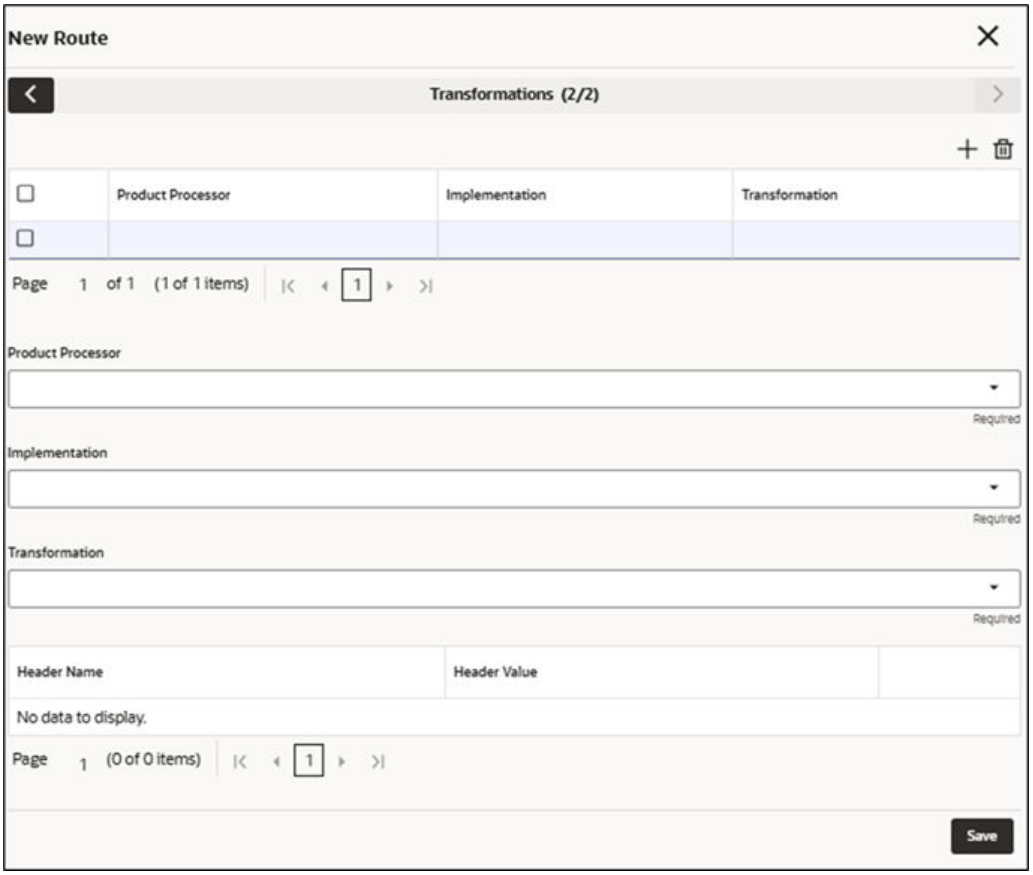
Transformations

Users can set a series of transformations for each routing to determine how a request is handled. The order of transformations in the list can be modified using a drag-and-drop feature.

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

      The **Transformations** screen is displayed.

Figure 8-4 Transformations



b. Specify the fields on **Transformations** screen.

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

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

Table 8-3 Transformations - 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.

c. Specify the header values if required.

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

Edit Route

The user can modify the routing details.

7. On **Routing** screen, click **Edit** icon.

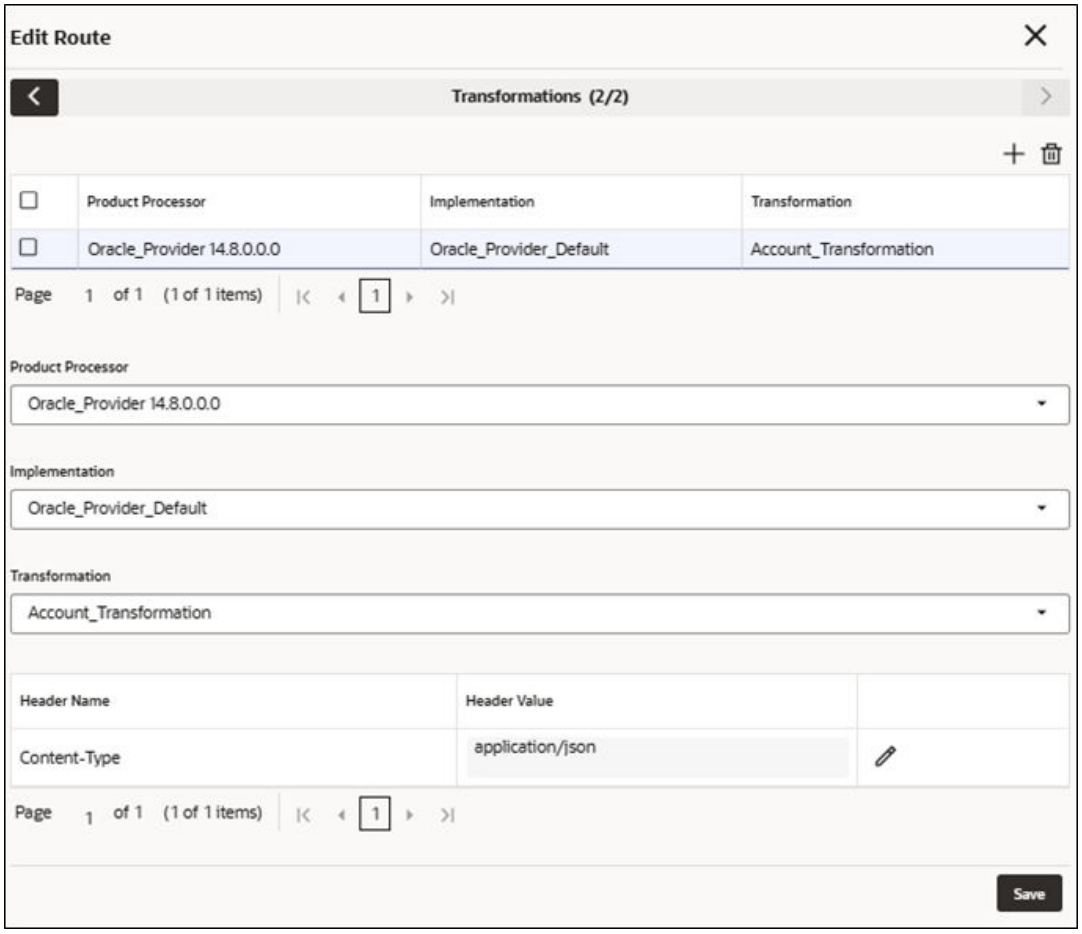
   The **Edit Route** screen is displayed.

Figure 8-5 Edit Route

8. Click **Next Step**.

   The **Edit Route - Transformations** screen is displayed.

Figure 8-6 Edit Route - Transformations



9. Click **Save** to save the modified transformation details.

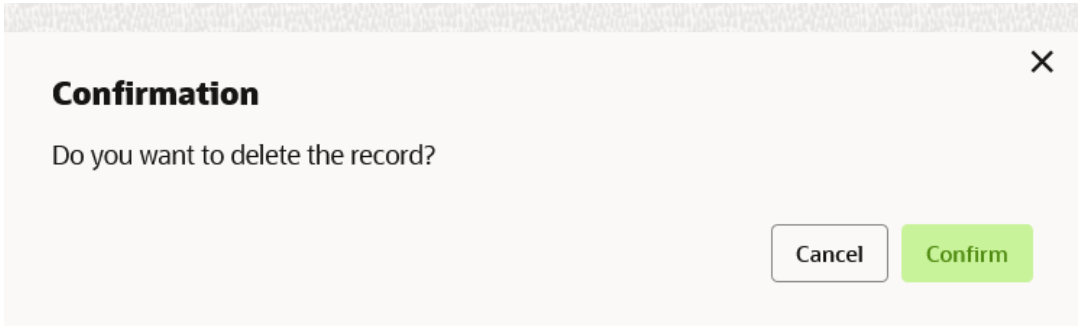
Delete Route

The user can delete the routing details.

10. On **Routing** screen, click **Delete**.

    The **Confirmation** screen is displayed.

Figure 8-7 Confirmation - Delete



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

Configuration

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

    The **Configuration** screen is displayed.

> **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 is displayed.

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

9 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 9-1 Chaining

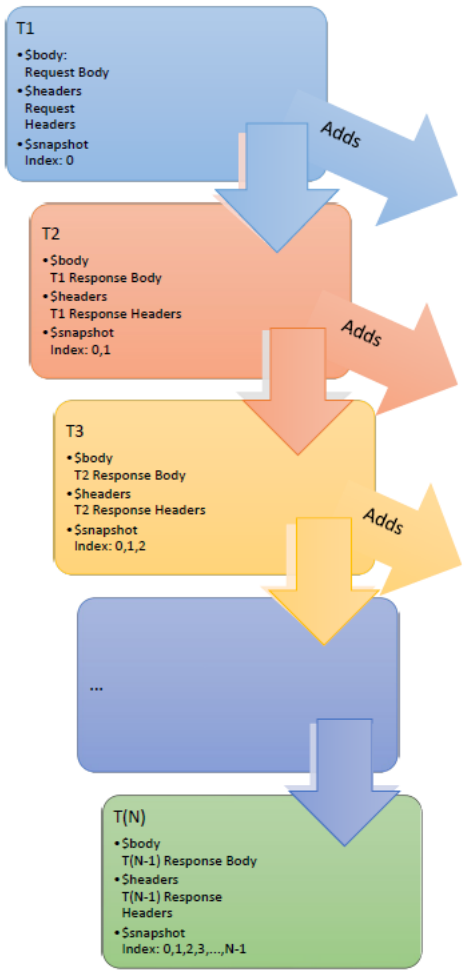


Table 9-1 Snapshot List

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

10

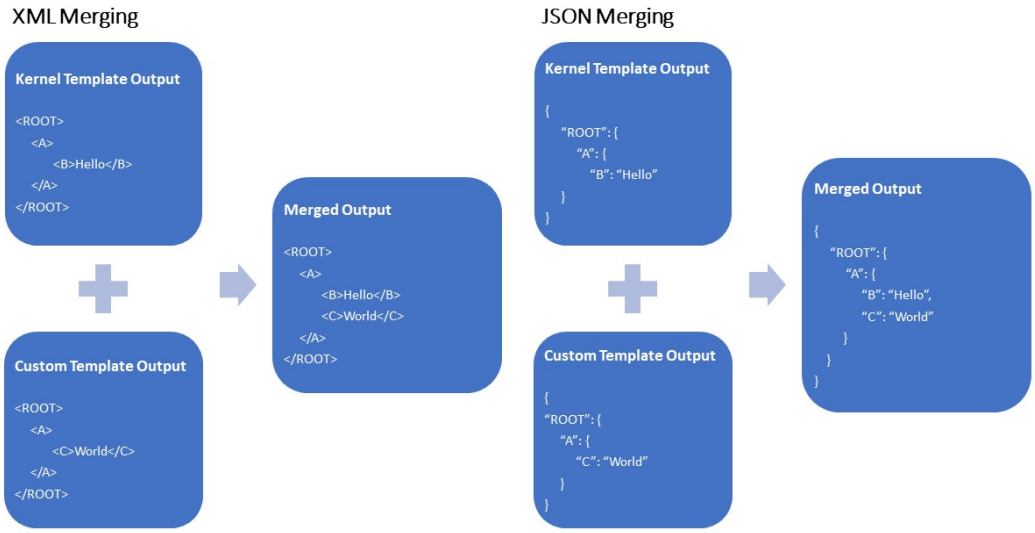
Template Extensibility

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 10-1 Extensibility - Example



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

- XML merging attributes

10.1 XML merging attributes

This topic contains the following subtopics:

- Identity Matcher
- Skip Matcher

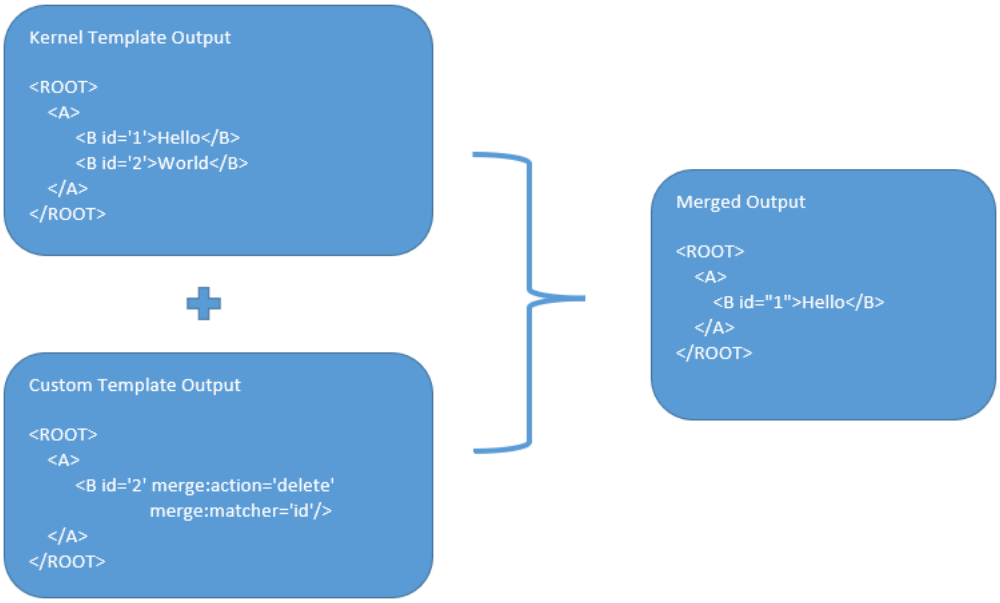
- Override Action
- Complete Action
- Replace Action
- Preserve Action
- Delete Action

10.1.1 Identity Matcher

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

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

Figure 10-2 Identity Matcher

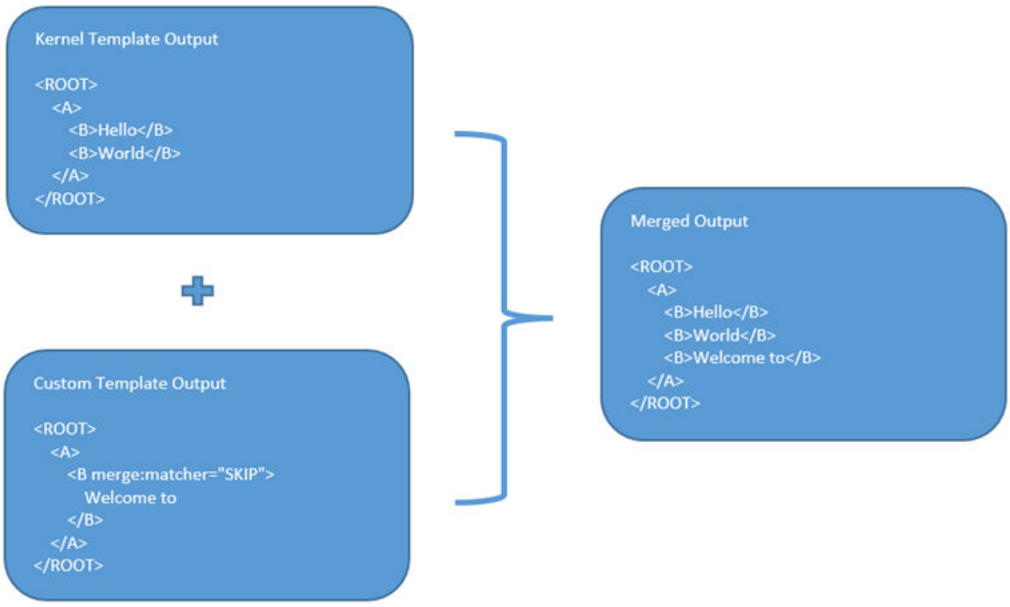


10.1.2 Skip Matcher

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

Syntax: `merge:action='SKIP'`

Figure 10-3 Skip Matcher

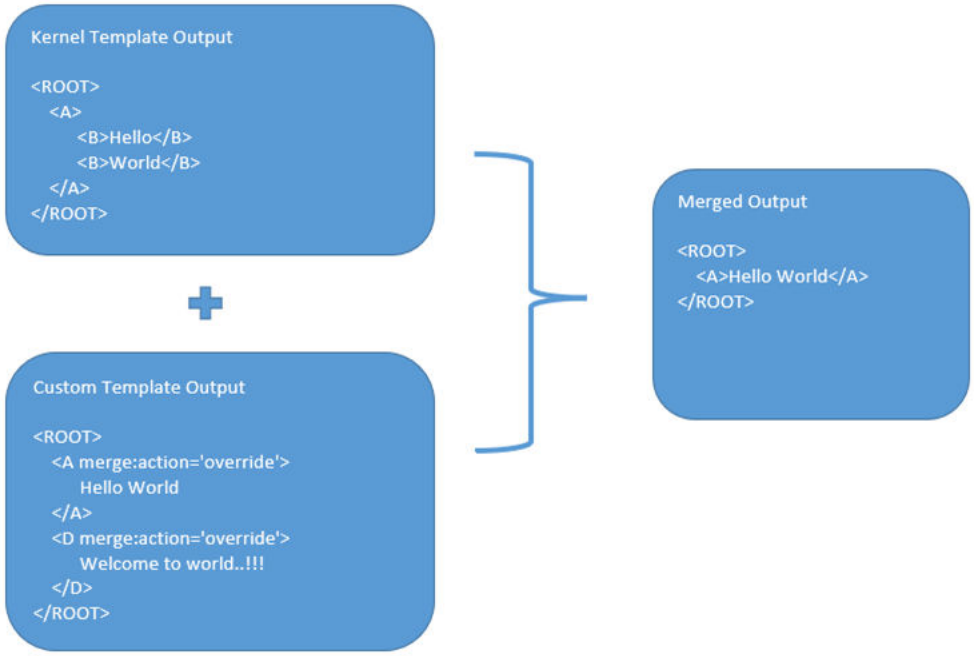


10.1.3 Override Action

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

Syntax: `merge:action='override'`

Figure 10-4 Override Action

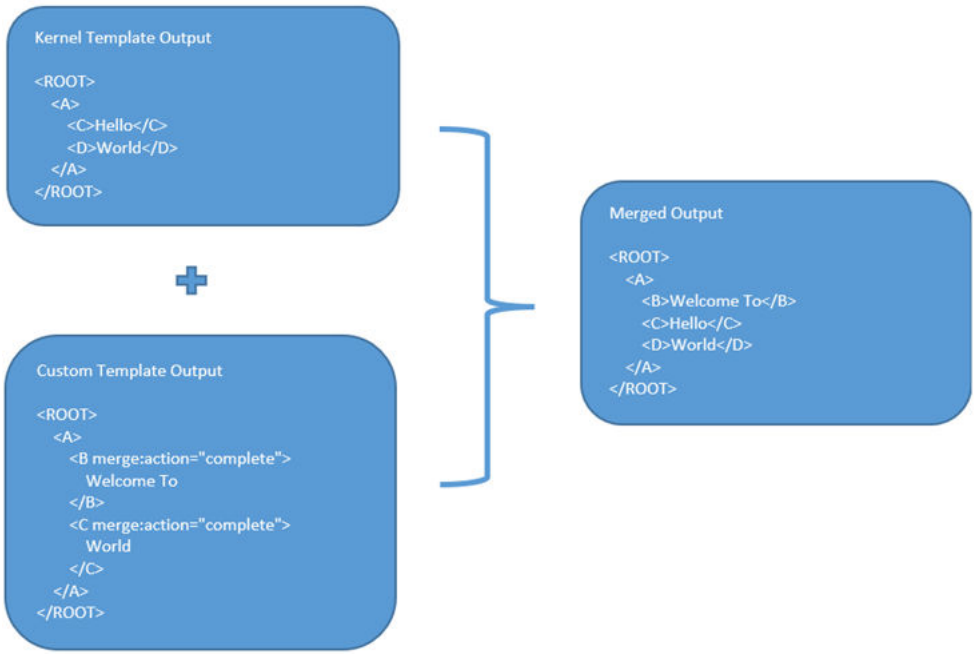


10.1.4 Complete Action

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

Syntax: `merge:action='complete'`

Figure 10-5 Complete Action

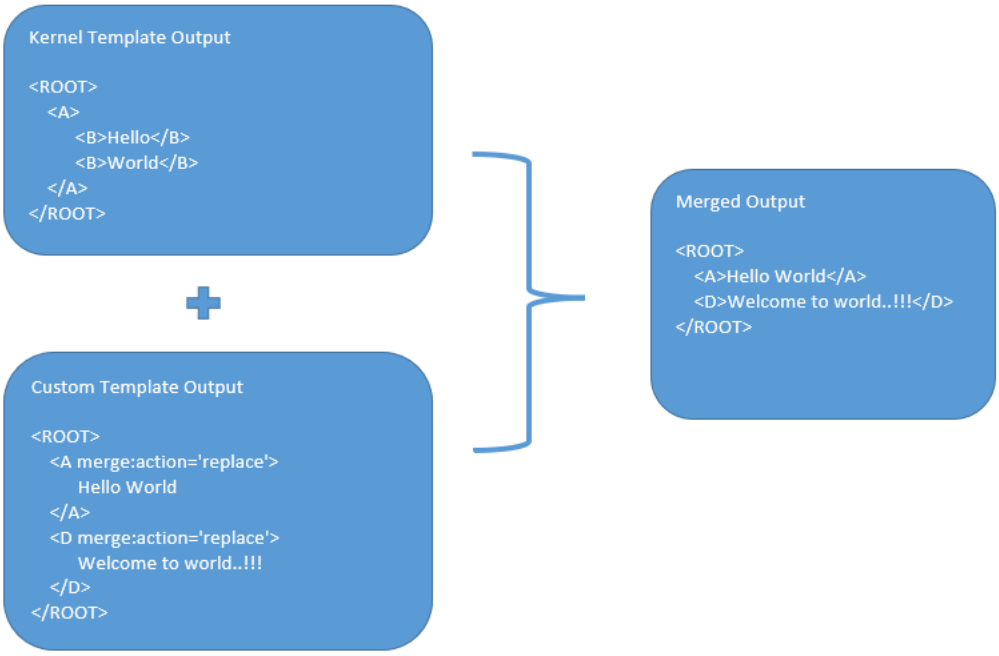


10.1.5 Replace Action

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 10-6 Replace Action

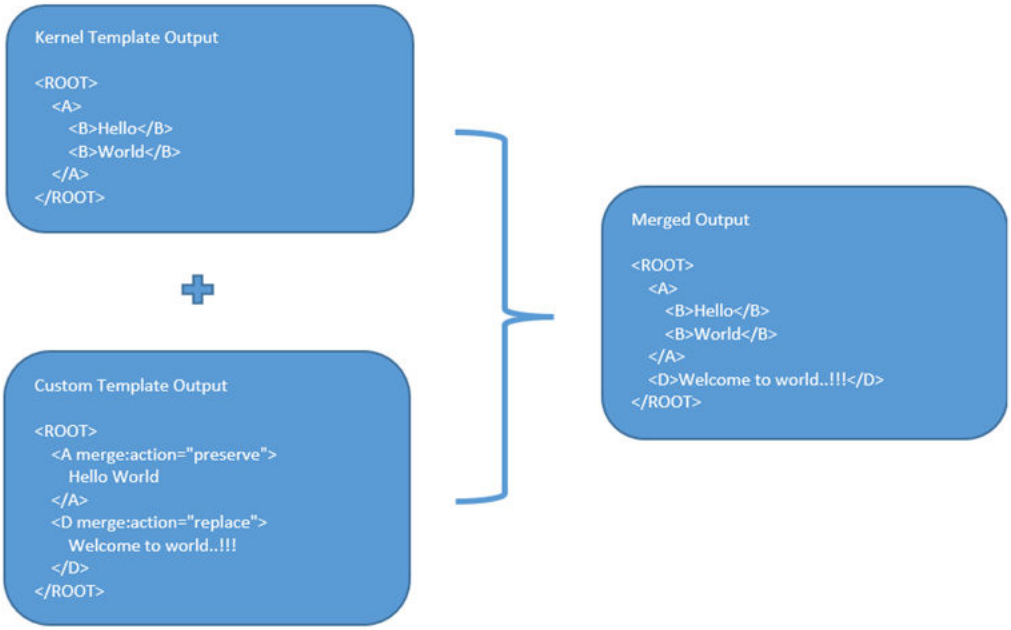


10.1.6 Preserve Action

No replace action is performed on the original element.

Syntax: `merge:action='preserve'`

Figure 10-7 Preserver Action

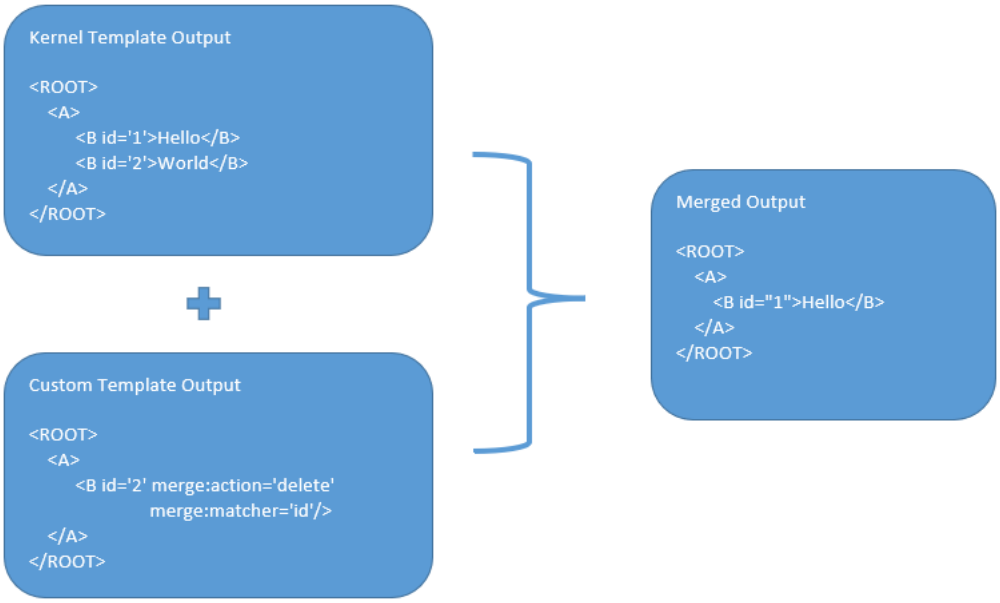


10.1.7 Delete Action

Deletes the original element.

Syntax: `merge:action='delete'`

Figure 10-8 Delete Action



11

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.

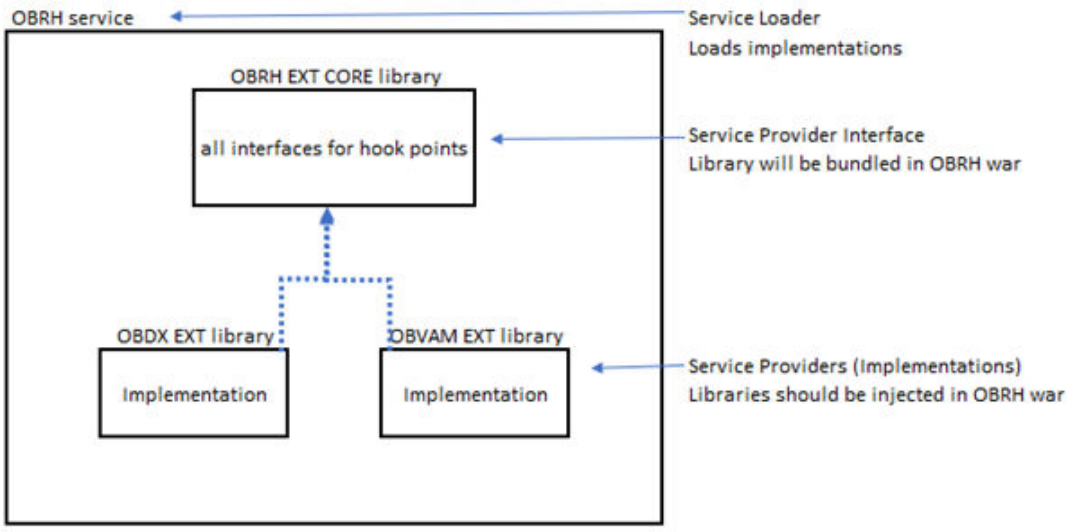
In order to resolve table space issue of Audit table (CMC_RH_AUDIT_EVENT_LOG), Database Management Team has to configure database job which should be triggered after routingHubAuditRetentionJob. This database job can be redefining the table (DBMS_REDEFINITION) after purging/archiving is done or other 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.

12
Hookpoints

Service Provider Interface (SPI) mechanism is used to make Routing Hub more extensible. SPI provides an option to extend interfaces without modifying the core application. All we need to do is provide a new implementation of the service that follows certain rules and plug it into the application.

Using the SPI mechanism, the application will load the new implementation and work with it.

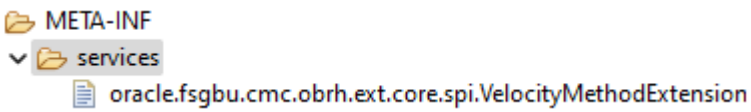
Figure 12-1 Hookpoints



Below steps to follow for specifying implementation:

- Extract the "cmc-obrh-ext-core-x.y.z.jar" (Extension Core) library from "cmc-obrh-service-x.y.z.war" artifact.
- Create library by consuming extension core library of Routing Hub.
- Specify the required implementations.
- In order to get it discovered, provider configuration file has to be created under "META-INF" as below:

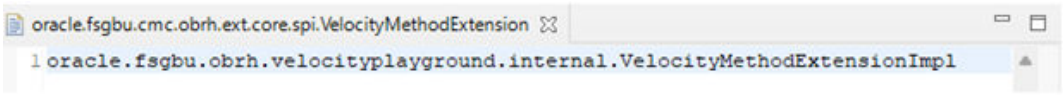
Figure 12-2 META-INF



Here, file name will be "PackageName.InterfaceName".

- Specify the implementation in above file as below:

Figure 12-3 Implementation



Here, file name will be "PackageName.InterfaceName".

- Inject the implementation library in "cmc-obrh-service-x.y.z.war" artifact.
- Available Interfaces

12.1 Available Interfaces

VelocityMethodExtension

This interface is for using existing client-specific logic in velocity templates.

In order to use it in velocity templates, processInput method needs to be called.

Syntax: $custom.processInput(String implementationName, Object... args)

Here, implementationName is mandatory if multiple implementations are present.

SoapOutInterceptorExtension

This interface is for processing the unmarshalled message data.

> **Note:**
>
> During PRE_LOGICAL and PRE_LOGICAL ENDING phases, the interceptions will be made automatically if the implementation is present.

SocketMessageHandlerExtension

This interface is for manipulating inbound and outbound socket messages.

Currently, supported socket implementations are Netty and JAVA socket and supported message types are string and hexstring.

13
Multipart Request

This topic provides the sample template for the multipart request

Example 13-1 Multipart Request

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

14

URL Encoded Request

This topic provides the sample template for url encoded request.

Example 14-1 URL Encoded Request

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

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

15
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 15-1 Configuration

3. On **Configuration** screen, specify the fields.

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

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

Table 15-1 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 required number of calls is 10, you need to record at least 10 calls before you can determine the failure rate. If only nine calls are logged, the circuit breaker will not switch to open, even if all nine calls are unsuccessful.
Failure rate threshold	Specify the failure rate threshold in percentage. If the failure rate meets or exceeds the threshold, the breaker opens and begins to short-circuit 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.
Allow data masking	Turn on the toggle to hide sensitive information in request audit messages.
Regex patterns	Specify the regex patterns for identification of sensitive fields. **Note:** You can group values by using a sub-pattern that is placed inside parentheses ().

Example:

Table 15-2 Configuration - Field Entry Values

Field	Entry Values
Window Type	Count
Window Size	20
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.

16

Request Audit - Log

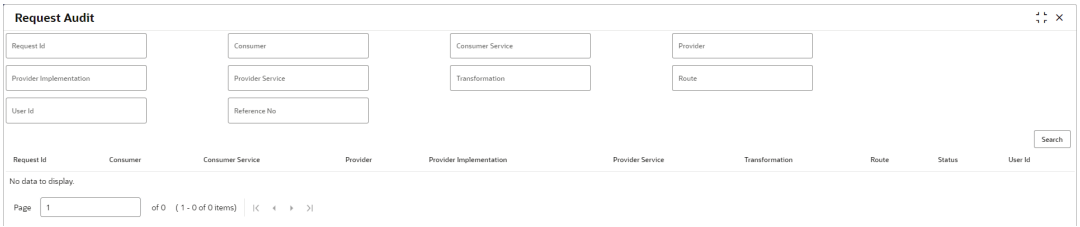
This topic describes the systematic instructions to check the audit log in Oracle Banking Routing Hub.

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 16-1 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 16-1 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 16-1 (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
Status	Status field indicates the outcome of the routing hub request with values indicating **SUCCESS**, **FAILURE**, or **PENDING**. **Note:** • **SUCCESS** signifies that the request was completed successfully. • **FAILURE** signifies that the request was unsuccessful. • **PENDING** signifies that the request is being processed.

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 16-2 Request Audit Details

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

Table 16-2 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.

17

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:

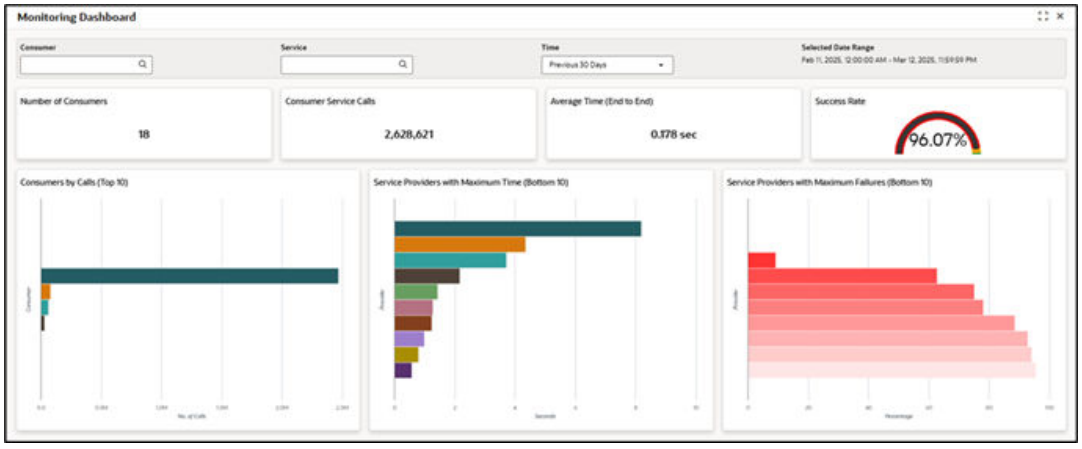
1. On **Home** screen, click **Task Management**. Under **Task Management** menu, click **Configure Tasks**.
2. Select **Schedule** option.
3. Select **Task Name** as routingHubAuditRetentionJob and **Task Trigger Name** will be generated automatically.
4. Specify the CRON expression to daily EOD.

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 not be 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 17-1 Monitoring Dashboard

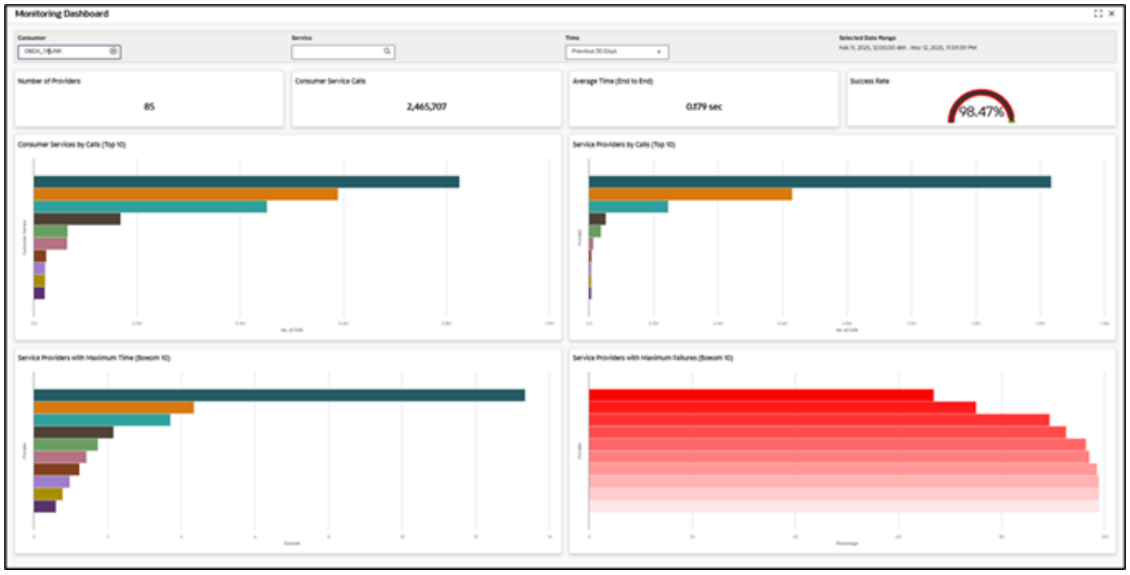


- **Number of Consumers:** This widget displays total number of consumers configured in the Oracle Banking Routing Hub.
- **Consumer Service Calls:** This widget displays total number of consumer services requested during chosen period.
- **Average Time (End to End):** This widget displays the average 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 17-2 Consumer Page



This page displays following information:

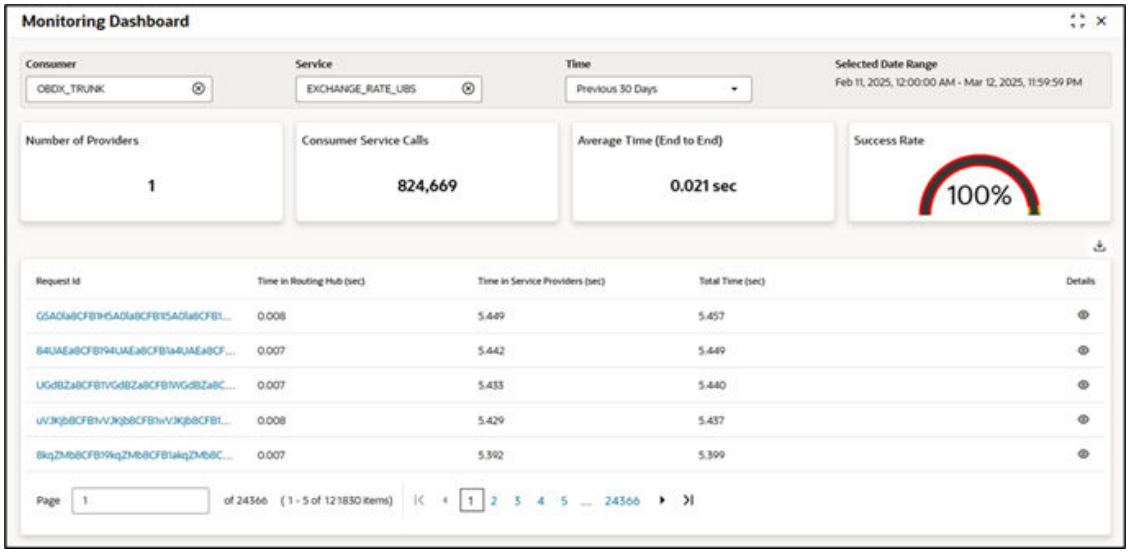
- **Number of Providers:** This widget displays the total number of service providers configured in Oracle Banking Routing Hub for the selected consumer.
- **Consumer Service Calls:** This widget displays total number of consumer services requested by the selected consumer during chosen period.

- **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 17-3 Consumer Service Page



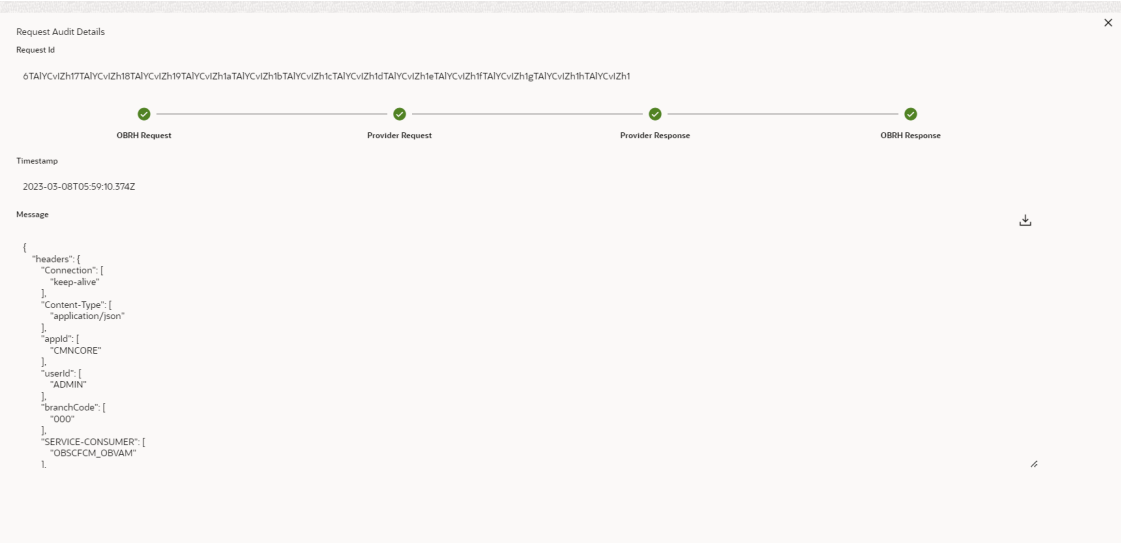
- **Number of Providers:** This widget displays total number of service providers to which this request is routed to complete the integration.
- **Consumer Service Calls:** This widget displays total number of consumer services made during chosen period.
- **Average Time (End to End):** This widget displays 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 17-4 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.
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.

18
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

– This method is for parsing JSON string
Syntax: $custom.parseJson(jsonString)

Parameters:

* jsonString - 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.

19

Oracle Banking Routing Hub VM Arguments

This topic provides information about Oracle Banking Routing Hub VM arguments.

Common Core Managed Server

Table 19-1 CMC-OBRH-SERVICE

Parameters	Default	Values
cmc-obrh-services.server.port	-	<SERVER_PORT>
cmc-obrh-services.server.port	-	<SERVER_PORT>
obrh.db.jndi	-	<CMNCORE_JNDI>
cmc-obrh-services.oic.secretStore.url	-	<OIC_SECRET_STORE_URL>
cmc-obrh-services.audit.retention.days	-	<AUDIT_RETENTION_POLICY_DAYS>
cmc-obrh-services.audit.retention.archival	-	Y / N (Y for archiving and N for purging)

Table 19-2 Enable and configure connection pooling for REST calls

Parameters	Default	Values
obrh.rest.connectionpool.enabled	false	true / false
obrh.rest.connectionpool.totalConnectionCount	20	<POOL_TOTAL_CONN_COUNT>
obrh.rest.connectionpool.maxConnectionCountPerRoute	2	<POOL_MAX_CONN_PER_ROUTE>
obrh.rest.connectionpool.timeToLive.ms	-1	<POOL_TTL>

Table 19-3 Receive routing failure mail notification via plato-alerts-management-service

Parameter	Default	Values
obrh.alerts.enabled	false	true / false

Table 19-4 Change approach for auditing

Parameters	Default	Values
obrh.audit.type	KAFKA	DEFAULT / KAFKA / JMS / LOG / OFF For KAFKA option, cmc-obrh-kafka-consumer service needs to be deployed. For JMS option, cmc-obrh-jms-consumer service needs to be deployed.

Table 19-4 (Cont.) Change approach for auditing

Parameters	Default	Values
obrh.audit.type.log.event	NONE	DISPATCH_REQUEST / DISPATCH_RESPONSE / ROUTE_INVOKE_START / ROUTE_INVOKE_FAILURE / TRANSFORMATION_TEMPLATE_EVALUATION_START / TRANSFORMATION_TEMPLATE_EVALUATION_END / TRANSFORMATION_EXTENDED_TEMPLATE_EVALUATION_START / TRANSFORMATION_EXTENDED_TEMPLATE_EVALUATION_END / PROVIDED_SERVICE_REQUEST / PROVIDED_SERVICE_RESPONSE This property is used to specify the events (comma-separated values) for which CLOB data needs to be logged and only considered if obrh.audit.type is LOG

Table 19-5 Overwrite the customization that is not part of configuration json

Parameters	Default	Values
obrh.import.overwrite	false	true / false

Table 19-6 Use Custom Keystore and Truststore for HTTPS scheme

Parameters	Default	Values
obrh.keystore.password.encoded	-	true / false (true, if password is base 64 encoded)
obrh.truststore.path	-	<TRUSTSTORE_PATH>
obrh.truststore.password	-	<TRUSTSTORE_PASSWORD>
obrh.usekeystore	-	true / false (true, if keystore is required along with truststore)
obrh.keystore.path	-	<KEYSTORE_PATH>
obrh.keystore.password	-	<KEYSTORE_PASSWORD>
obrh.keystore.alias	-	<KEYSTORE_ALIAS_LIST>
obrh.keystore.aliaspassword	-	<KEYSTORE_ALIAS_PASSWORD_LIST>
obrh.ssl.protocol	TLS	TLS / TLSv1 / TLSv1.1 / TLSv1.2

Table 19-7 For tomcat deployment

Parameters	Default	Values
obrh.server.isJavaEE	true	true / false (false for tomcat)

Table 19-7 (Cont.) For tomcat deployment

Parameters	Default	Values
obrh.taskexecutor.corepoolsize	50	<CORE_POOLSIZE>
obrh.taskexecutor.maxpoolsize	50	<MAX_POOLSIZE>
obrh.taskexecutor.queuecapacity	100	<QUEUE_CAPACITY>

Set Proxy settings for HTTPS: As per the Java Networking documentation, HTTPS protocol handler will use the same as the http handler (i.e. http.nonProxyHosts). But in case of Weblogic, http.nonProxyHosts will not work for some reason. So, use https non proxy host argument (i.e. https.nonProxyHosts).

Table 19-8 Set Proxy settings for HTTPS

Parameters	Default	Values
https.proxyHost	-	<PROXY_HOST_NAME>
https.proxyPort	-	<PROXY_PORT>
https.nonProxyHosts	-	<NON_PROXY_HOST_LIST>
http.nonProxyHosts	-	<NON_PROXY_HOST_LIST>

This property will enforce WebLogic Server to use SUN SSL implementation (javax package) rather than the WebLogic one.

Table 19-9 Support SSL based SOAP provider calls in weblogic environment

Parameters	Default	Values
UseSunHttpHandler	-	true

Table 19-10 CMC-OBRH-KAFKA-CONSUMER

Parameters	Default	Values
cmc-obrh-kafka-consumer.server.port	-	<SERVER_PORT>
obrh.audit.id-generator	UUID	UUID / SNOWFLAKE

Table 19-11 CMC-OBRH-JMS-CONSUMER

Parameters	Default	Values
cmc-obrh-jms-consumer.server.port	-	<SERVER_PORT>
cmc-obrh-jms-consumer.connectionFactory	-	<JMS_CONN_FACTORY_JNDI>
cmc-obrh-jms-consumer.queue	-	<JMS_CONN_QUEUE_JNDI>

Table 19-12 Change ID generator

Parameters	Default	Values
obrh.audit.id-generator	UUID	UUID / SNOWFLAKE

Plato Core Managed Server

Oracle Banking Routing Hub is using Multipart for Import feature. By default, spring supports max 1MB file size and 10MB request size for Multipart.

To import bigger files,

`plato-api-gateway.multipart.max-file-size=<MAX_FILE_SIZE>` (default is 200MB)
`plato-api-gateway.multipart.max-request-size=<MAX_REQUEST_SIZE>` (default is 200MB)

> **Note:**
>
> -1 for no size constraint
> **Example**,
>
> plato-api-gateway.multipart.max-file-size=-1
> plato-api-gateway.multipart.max-request-size=-1

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

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

Screen Name	Functional Activity Codes	Action	Description
Routing Hub	CMC_FA_RH_SERVICECONSUMER_CREATE	CREATE	Creates consumer
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

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

Screen Name	Functional Activity Codes	Action	Description
Routing Hub	CMC_FA_RH_SERVICECONSUMER_SERVICE_EXPORT	EXPORT	Exports service
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

Index