# Routing Hub Configuration User Guide
# Oracle Banking Electronic Data Exchange for Corporates
# Release 14.7.0.0.0

Part No. F73521-01

November 2022

Routing Hub Configuration User Guide

November 2022

Oracle Financial Services Software Limited

Oracle Park

Off Western Express Highway

Goregaon (East)

Mumbai, Maharashtra 400 063

India

Worldwide Inquiries:

Phone: +91 22 6718 3000

Fax:+91 22 6718 3001

www.oracle.com/financialservices/

# Table of Contents

# 1. Preface

## 1.1 Purpose

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

## 1.2 Intended Audience

This document is intended for the following audience:

- Customers
- Partners

## 1.3 Document 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.

## 1.4 Access to Oracle Support

Oracle customers 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.

## 1.5 Structure

This manual is organized into the following categories:

Preface gives information on the intended audience. It also describes the overall structure of the User Manual.

The subsequent chapters describe following details:

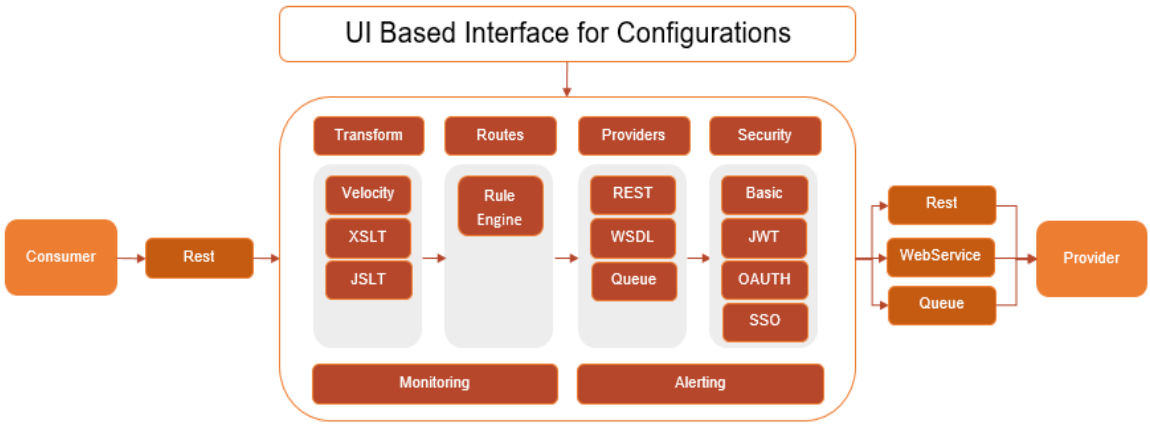
- Introduction
- Preferences & Database
- Configuration / Installation

# 2. Introduction

Oracle Banking Routing Hub is routing hub which enables seamless & standardized integrations between FSGBU Banking Product using configurations provided as part of the product Infrastructure.
Consumer Application does not need to know

- Servicing Providers or Product Processors - Product processor to which the integration is required.
- Name of the Service - Logical name of the service e.g. Funds Transfer, Letter of Credit Initiation
- Messaging structure of Service - Structure of the message e.g JSON, XML, multipart request.
- Communication Protocol – Web services, Rest API, Queue.
- Can be integrated with different versions of a Product processors

In this document we have shown the maintenance of two product i.e." Oracle Service Consumer" as Service Consumer and "External Product Processor" as Service Provider




## 2.1 Acronyms & Definitions

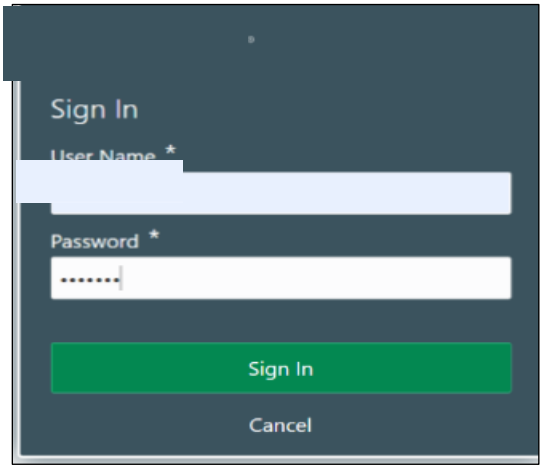
Following are some of the acronyms and abbreviations you are likely to find in this user manual:

| Abbreviation & Definitions | Description |
|---|---|
| Service Consumers | Are Applications need to integrate with multiple product processors with loosely couple integration layer available as Oracle Banking Routing Hub. Service Consumer integrates with Oracle Banking Routing Hub |

| | |
|---|---|
| Service Providers | This are product processors available to serve the request send by Oracle Banking Routing Hub on behalf of Service Consumer |
| Service | Are Soap Web Services imported through WSDL or Rest Web Services imported through Swagger |
| Headers | Are headers require by product processor while sending request |

# 3. Start Maintenance - Login Screen

Open Browser, Hit URL and Launch **Oracle Banking Routing Hub**.




| Component briefing | | | | |
|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Comments** |
| **User Name** | Text Box | Yes | Alphanumeric | |
| **Password** | Text Box | Yes | Alphanumeric with special characters | |
| **Sign In** | Button | | | Navigates to Dashboard / menu screen |
| **Cancel** | Button | | | |

# 4. Main Menu Screen

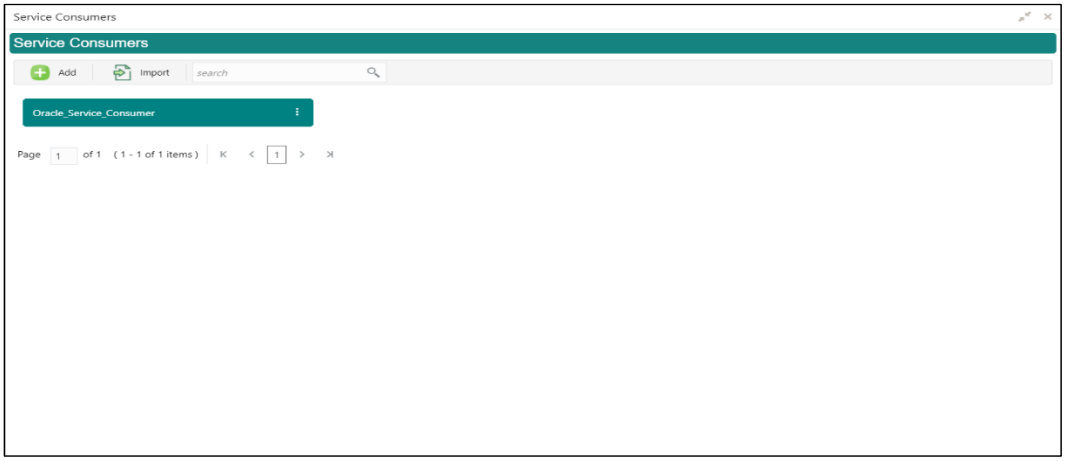

| Component briefing | | |
|---|---|---|
| **Component Name** | **Component Type** | **Comments** |
| **Core Maintenance** | Main menu item | |
| **Routing Hub** | Sub menu item | |
| **Configuration** | Sub menu option | Navigates to Configuration screen |
| **Service Consumer** | Sub menu option | Navigates to Service Consumer screen |
| **Request Audit** | Sub menu option | Navigates to Request Audit screen |

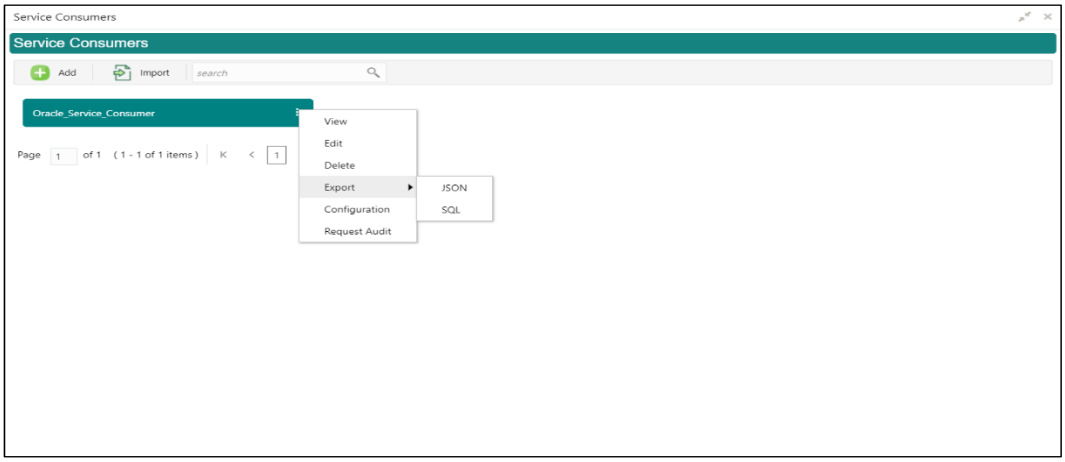
# 5. Service Consumer

Service Consumer is Oracle product which will invoke Oracle Banking Routing Hub API, Oracle Banking Routing Hub will analyse, evaluate destination product processor and transform data into format of the same.

Service Consumer comprises of the source and destination integration details.

Navigation: **Core Maintenance -> Routing Hub -> Service Consumers**



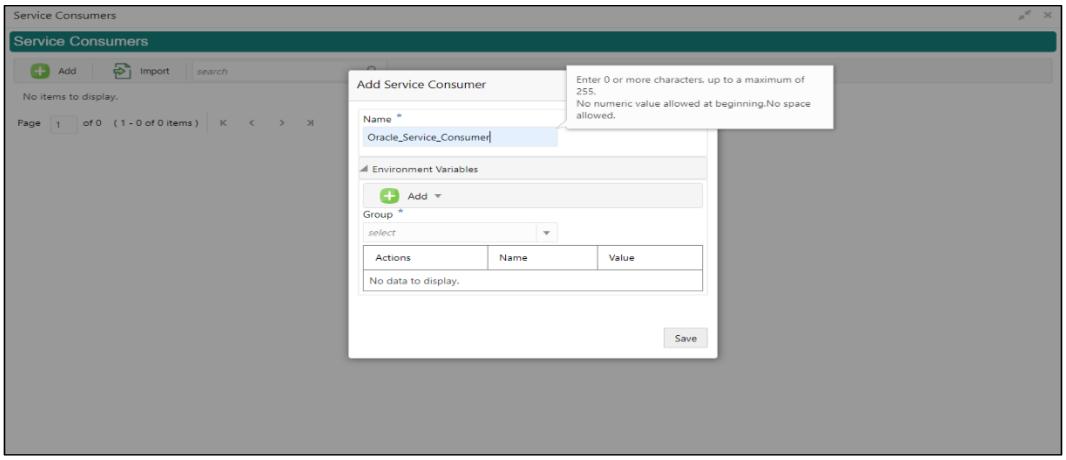

| **Component briefing** | | | |
|---|---|---|---|
| **Component Name** | **Component Type** | **Condition** | **Comments** |
| **Add** | Button | | Pops up add dialog |
| **Import** | Button | | Pops up import dialog |

| | | | |
|---|---|---|---|
| **Search** | Combo Box One | | Provides search functionality with case insensitive (Service Consumer Name) |
| **Navigation: Service Consumers -> 3 dot icon (operation menu)** | | | |
| **View** | menu option | Non-editable | Pops up view dialog |
| **Edit** | menu option | | Pops up edit dialog |
| **Delete** | menu option | | |
| **Export** | Sub menu item | | |
| **JSON** | menu option | | Exports in JSON |
| **SQL** | menu option | | Exports in SQL |
| **Configuration** | menu option | | Pops up configuration dialog |
| **Request Audit** | menu option | | Pops up request audit log |

## 5.1 Add

User can create Service Consumer manually.

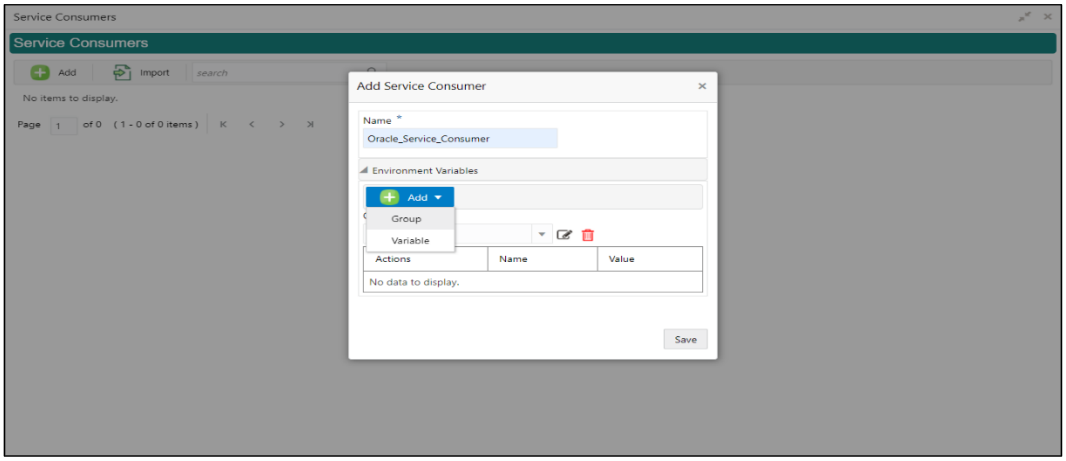
Navigation: **Service Consumers -> Add**

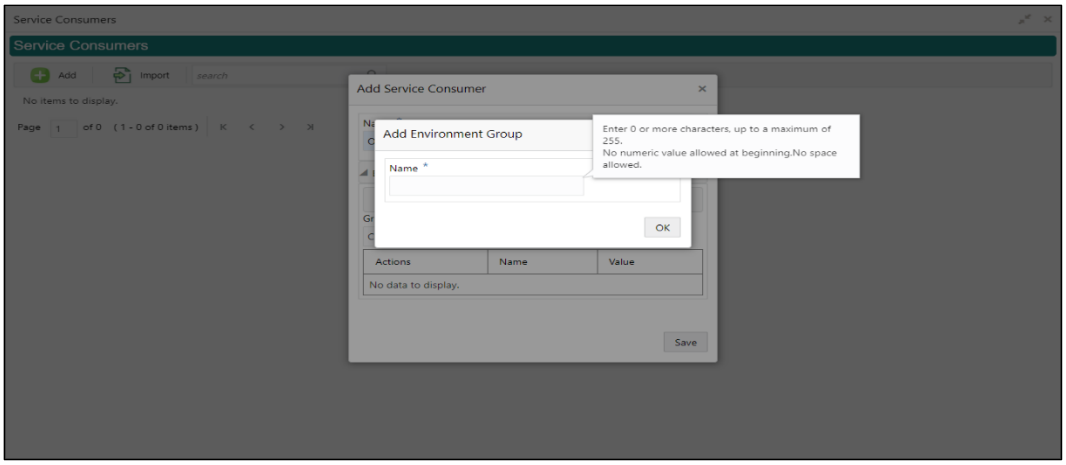


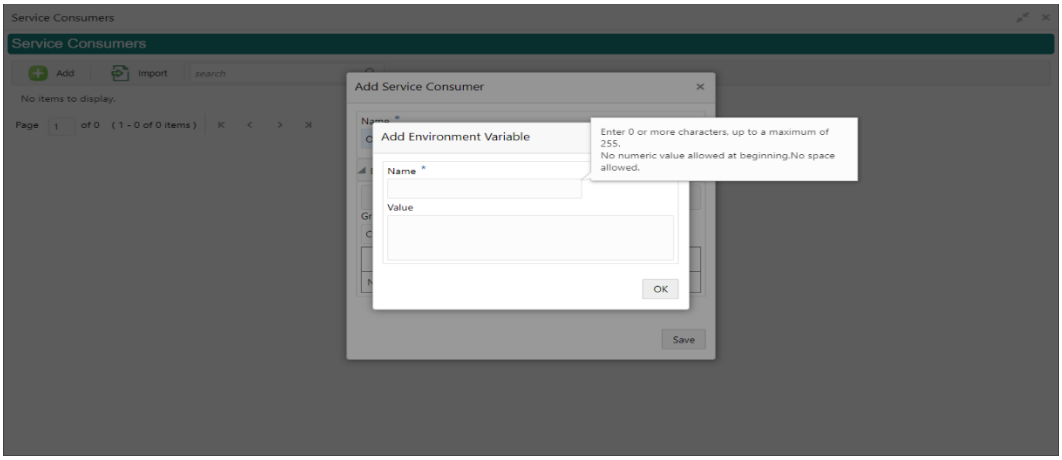
| Component briefing | | | | | |
|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Validation** | **Comments** |
| **Name** | Text Box | Yes | Alphanumeric with special characters | • Name cannot be blank<br>• Enter 0 or more characters, up to a maximum of 255<br>• No numeric value at beginning and no space allowed | Unique Service Consumer name |
| **Environment Variables** | Table Content | | | | |
| **Save** | Button | | | | Saves the Service Consumer |

### 5.1.1 Environment Variables

User will be able to define the group of variables which can be accessed throughout the specific consumer's configuration.
Below is the syntax for accessing environment variables:
$env.Environment_Group_Name.Environment_Variable_Name
eg: $env.COMMON.BRANCH_CODE





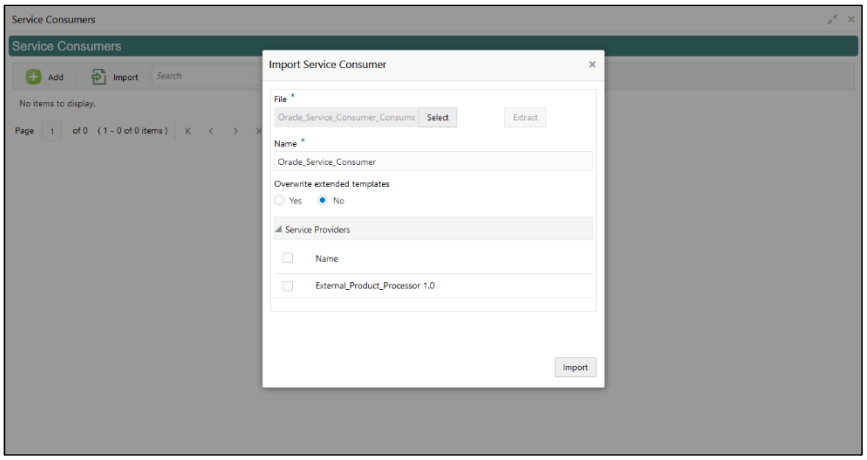




| **Component briefing** | | | | | |
|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Validation** | **Comments** |
| **Add** | Menu Item | | | | |
| **Group** | Menu option | | | | Pops up add group dialog |
| **Variable** | Menu option | | | | Pops up add variable dialog |
| **Navigation: Service Consumer -> Environment Variables -> 3 dot icon (operation menu)** | | | | | |
| **Edit** | menu option / icon | | | | Pops up edit dialog |
| **Delete** | menu option / icon | | | | Deletes group / variable |

| Environment Group / Variable | | | | | |
|---|---|---|---|---|---|
| **Name** | Text Box | Yes | Alphanumeric with special characters | • Name cannot be blank<br>• Enter 0 or more characters, up to a maximum of 255<br>• No numeric value at beginning and no space allowed | |
| **Value** | Text Area | | | | Value can either be hardcoded or Velocity mapping. |
| **OK** | Button | | | | Saves the group / variable and displays it in the list |

## 5.2 Import

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. User can also import zip file in order to import all the configuration JSON files together. Navigation: **Service Consumers -> Import**



| Component briefing | | | | | | |
|---|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Validation** | **Condition** | **Comments** |
| **File** | File picker | Yes | | Allows only to select one file | Accepts JSON and ZIP file | Pops up file selection dialog box |
| **Extract** | Button | Yes | | | | Extracts Consumer Name and Service Provider list from JSON file only and displays it in the respective elements. |
| **Name** | Text Box | Yes | Alphanumeric with special characters | • Name cannot be blank<br>• Enter 0 or more characters, up to a maximum of 255<br>• No numeric value at beginning and no space allowed | Editable | Name is required only for JSON file |
| **Overwrite extended templates** | Radio Button | Yes | | | | Predefined Values:<br>Yes / No<br>Yes: This option is for overwriting the extended templates in configuration and No: This option is for retaining the existing extended templates in configuration. |
| **Service Provider** | Collapsible Header & Content | | | | | Displays the list of service providers that are present in JSON file only |

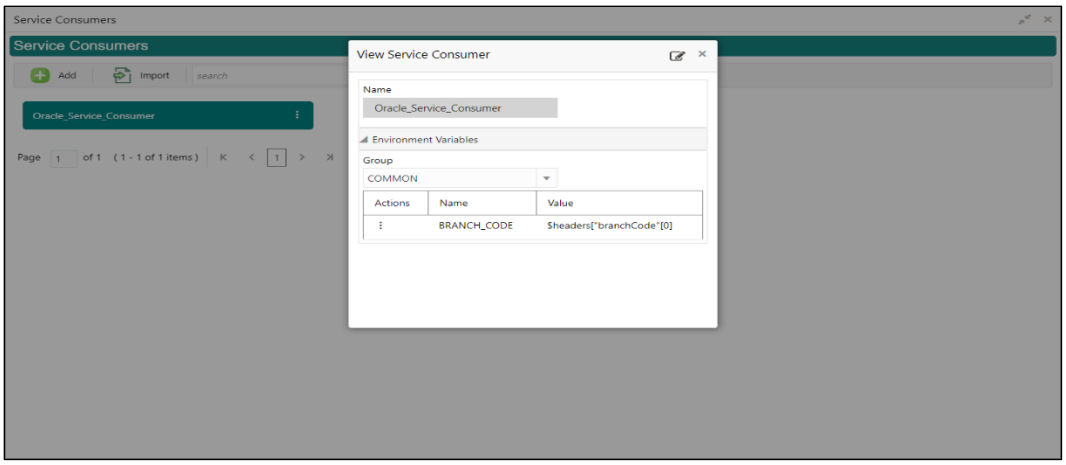
| | | | | | | |
|---|---|---|---|---|---|---|
| **Import** | Button | | | | | Imports Service Consumer |

**Below data needs to be changed after importing consumer configuration file:**

- Implementation Host and Port
- Implementation Authentication Password

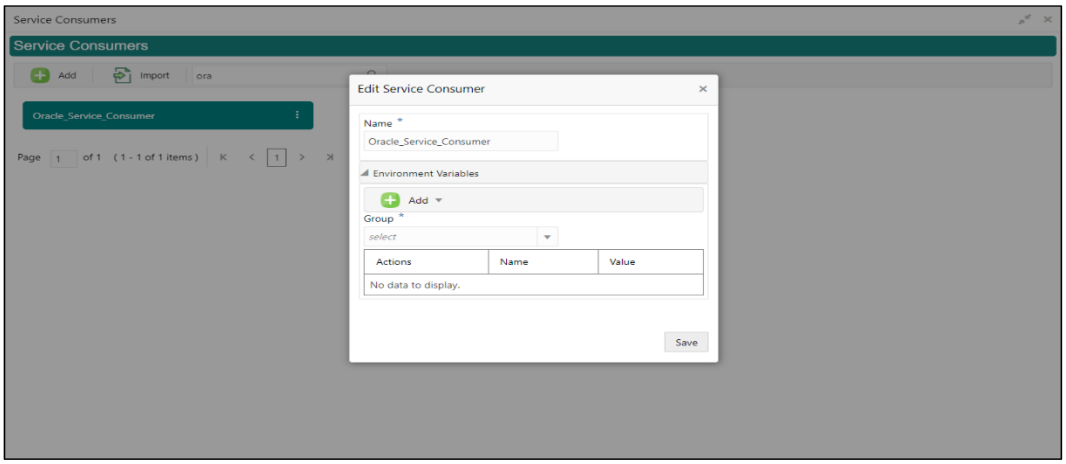
## 5.3 View

User can view consumer details and can also switch to edit form by clicking on edit icon.
Navigation: **Service Consumers -> Operation Menu (3 dot icon) -> View**



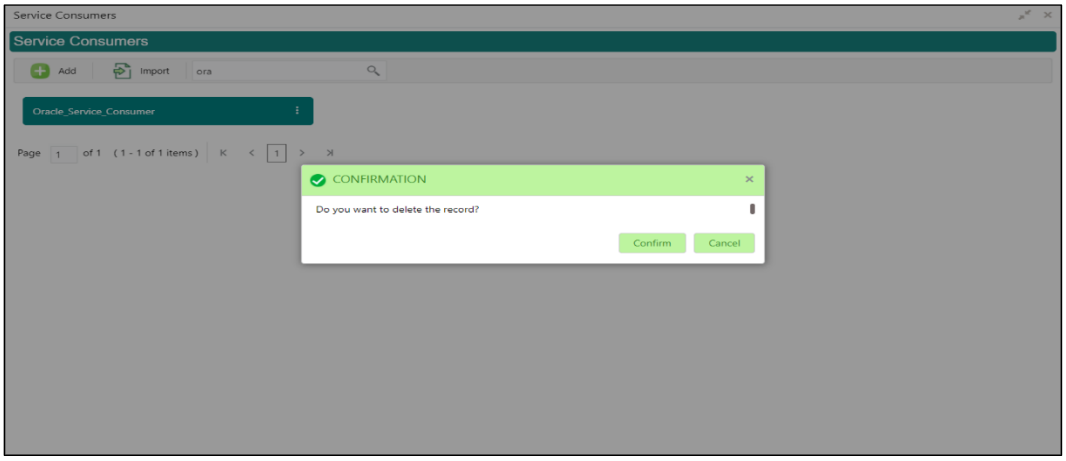

## 5.4 Edit

User can modify the consumer details.
Navigation: **Service Consumers -> Operation Menu (3 dot icon) -> Edit**



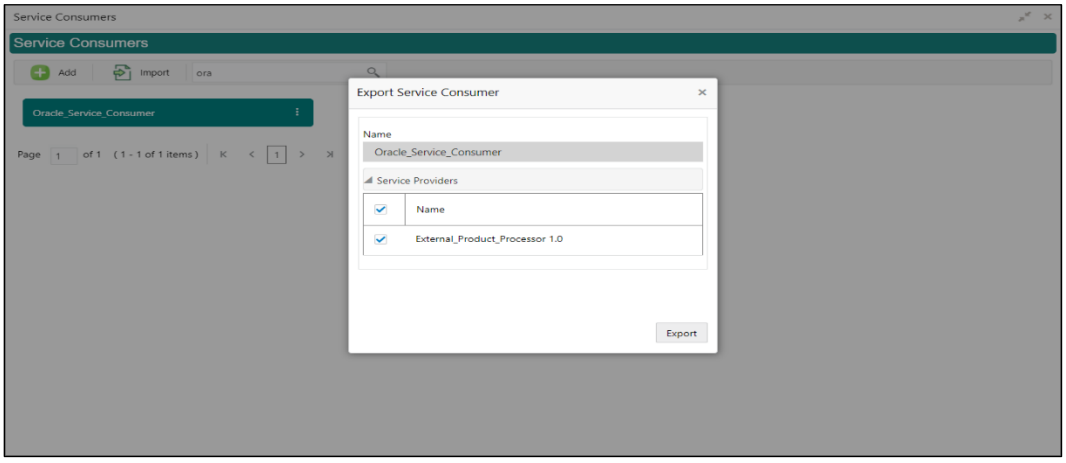
## 5.5 Delete

User can delete the consumer.
Navigation: **Service Consumers -> Operation Menu (3 dot icon) -> Delete**




## 5.6 JSON Export

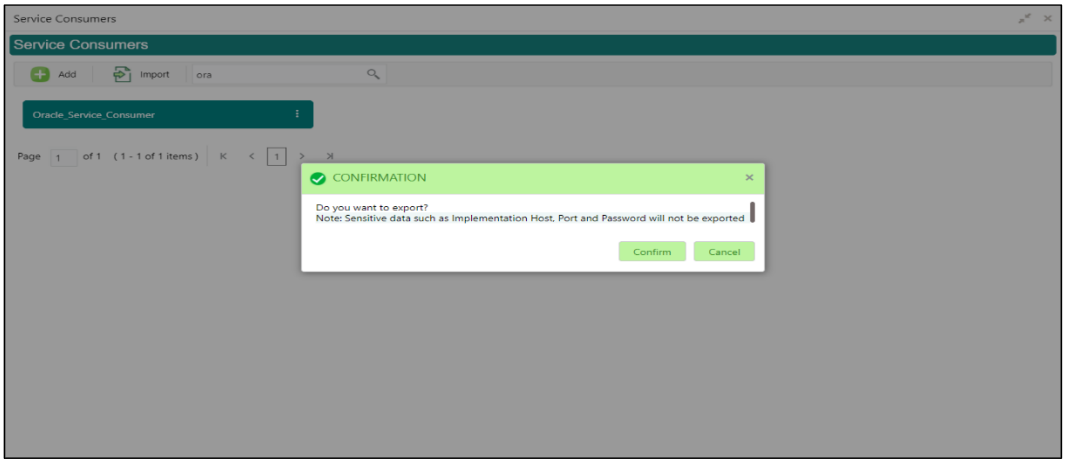
User can export the consumer configuration as JSON file.
Navigation: **Service Consumers -> Operation Menu (3 dot icon) -> Export -> JSON**




**Note:**
• User has an option to select service providers from the list which needs to be exported or can click on "Select All" option for all service providers.
• JSON Export feature will export 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

## 5.7 SQL Export

User can export the consumer configuration as SQL file.
Navigation: **Service Consumers -> Operation Menu (3 dot icon) -> Export -> SQL**

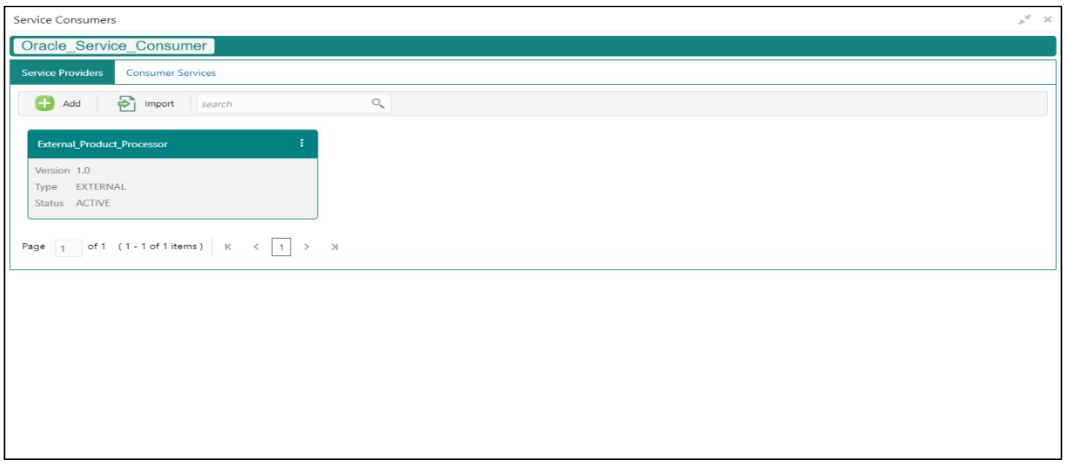


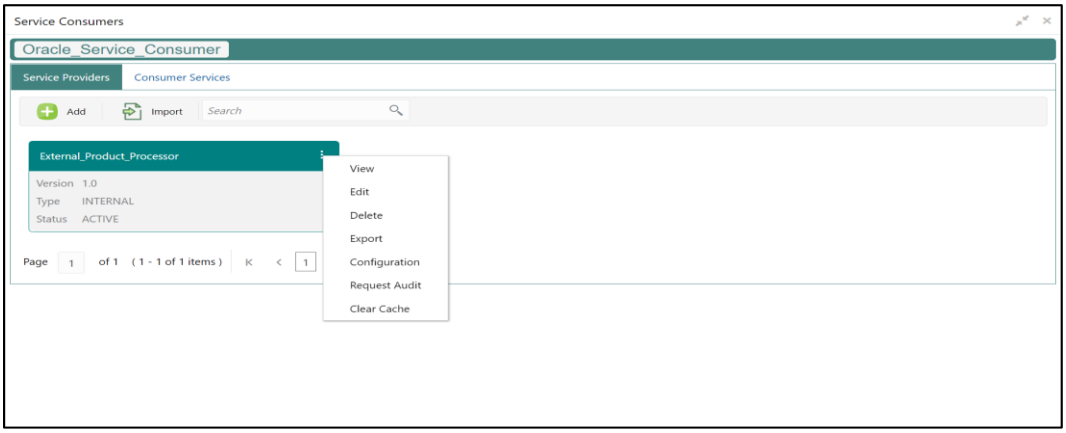

SQL Export feature will export entire configuration without Host, Port and Authentication Password details.

# 6. Service Providers

Service Providers are the product processors configure to process request send by Oracle Banking Routing Hub on behalf of service consumers.

Service Provider comprises of destination integration details. Navigation: **Core Maintenance -> Routing Hub -> Service Consumers -> <Specific Service Consumer> -> Service Providers**



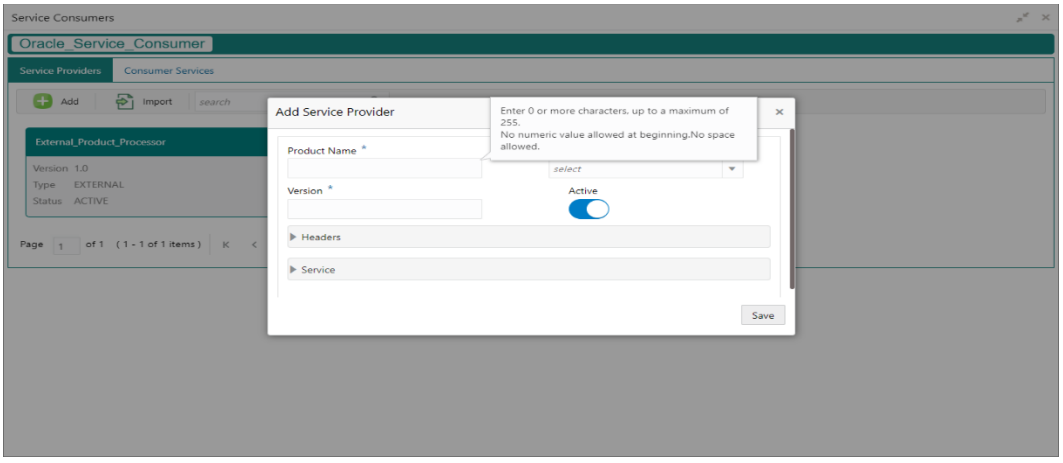




| **Component briefing** | | | |
|---|---|---|---|
| **Component Name** | **Component Type** | **Condition** | **Comments** |
| **<Service Consumer>** | Button | | Navigates back to Service Consumers |
| **Add** | Button | | Pops up add dialog |

| | | | |
|---|---|---|---|
| **Import** | Button | | Pops up import dialog |
| **Search** | Combo Box One | | Provides search functionality with case insensitive (Service Provider Name) |
| **Navigation: Service Providers -> 3 dot icon (operation menu)** | | | |
| **View** | menu option | Non-editable | Pops up view dialog |
| **Edit** | menu option | | Pops up edit dialog |
| **Delete** | menu option | | |
| **Export** | menu option | | Exports in JSON |
| **Configuration** | menu option | | Pops up configuration dialog |
| **Request Audit** | menu option | | Pops up request audit log |
| **Clear Cache** | menu option | | Clears SOAP client cache |

## 6.1 Add

User can create Service Provider manually.
Navigation: **Service Providers-> Add**

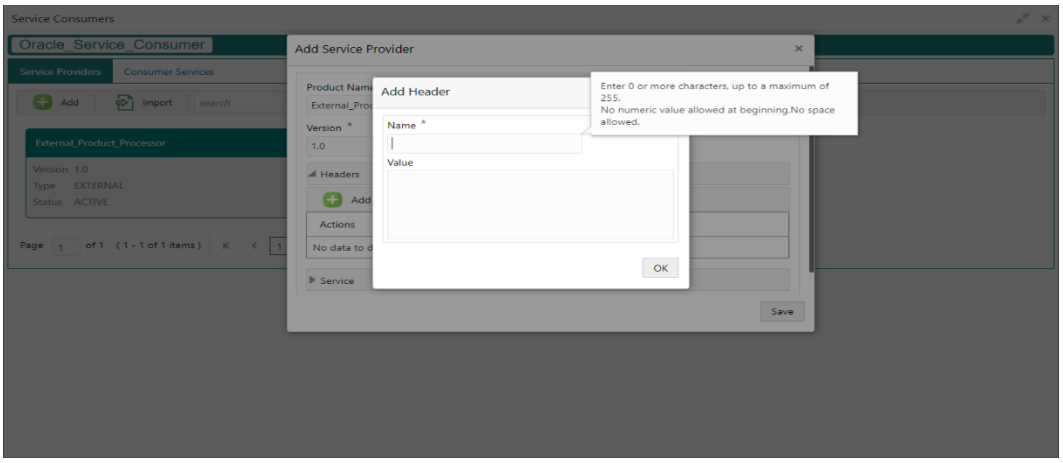


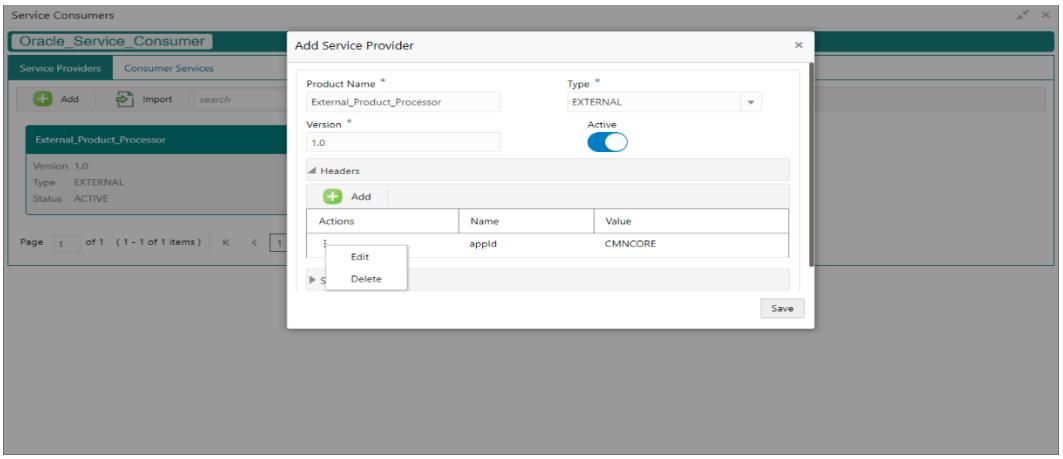
| Component briefing | | | | | |
|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Validation** | **Comments** |
| Product Name | Text Box | Yes | Alphanumeric with special characters | • Name cannot be blank<br>• Enter 0 or more characters, up to a maximum of 255.<br>• No numeric value at beginning and no space allowed. | Unique provider name |
| Type | Combo Box One | Yes | | | Predefined Values: INTERNAL / EXTERNAL<br>INTERNAL type should be used for oracle products.<br>EXTERNAL type should be used for non-oracle products |
| Version | Text Box | Yes | Number | • Version cannot be blank<br>• Enter 0 or more characters, up to a maximum of 255.<br>• Enter only numeric or decimal values. | Unique provider version |
| Active | Switch | | | | Predefined Values: ACTIVE / INACTIVE<br>If provider is marked as inactive, then all related routes will be stopped. |

| | | | | | |
|---|---|---|---|---|---|
| Headers | Collapsible Header & Content | | | | Provider specific headers |
| Service | Collapsible Header & Content | | | | Provider specific service details |
| Save | Button | | | | Saves the Service Provider |

## 6.1.1 Headers

External product processor might require some standard headers to be passed along with the request.
User can specify the headers which are required by service endpoints for its all implementations but not present in swagger file.
These headers can be configured in Oracle Banking Routing Hub using the steps given below

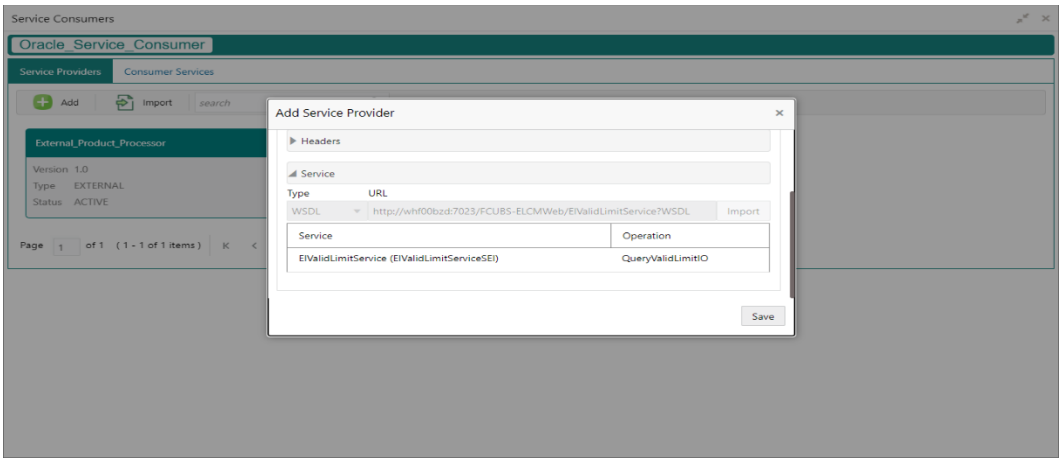



| Component briefing | | | | | |
|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Validation** | **Comments** |
| Add | Button | | | | Pops up add dialog |
| **Navigation: Service Providers -> Headers -> 3 dot icon (operation menu)** | | | | | |

| | | | | | |
|---|---|---|---|---|---|
| Edit | menu option | | | | Pops up edit dialog |
| Delete | menu option | | | | Deletes header |
| **Navigation: Service Providers -> Headers -> Add** | | | | | |
| Name | Text Box | Yes | Alphanumeric with special characters | • Name cannot be blank<br>• Enter 0 or more characters, up to a maximum of 255.<br>• No numeric value at beginning and no space allowed. | |
| Value | Text Area | Yes | Alphanumeric with special characters | • Enter 0 or more characters, up to a maximum of 255. | Value can either be hardcoded or can be Velocity mapping. |
| OK | Button | | | | Saves the header details and displays it in the list |

## 6.1.2 **Service**



| Component briefing | | | |
|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Comments** |
| Type | Combo Box One | Yes | Predefined Values: WSDL / SWAGGER / OTHERS |
| URL | Text Box | Yes | Service URL of the file location |
| Gateway Prefix | Text Box | | Gateway Prefix is context path of below formatted URL http://host:port/gateway-prefix/endpoint |
| Import | Button | | Extracts the service information from URL and displays it in the Service list |

#### 6.1.2.1 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.

In case there is a change in wsdl file, then same wsdl file need to be imported again in order to update the provided service information in routing hub.

#### 6.1.2.2 SWAGGER

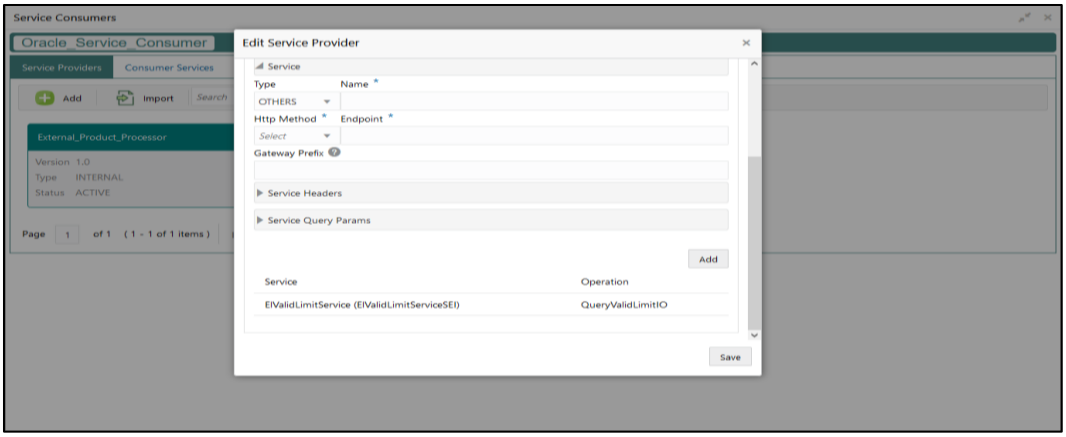
Swagger is an Interface Description Language for describing RESTful APIs expressed using JSON.

Currently, Swagger 2.0 & OpenAPI 3.0 both are supported.

In case 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.

#### *6.1.2.3* OTHERS

"OTHERS" option is for adding REST API details manually when provider does not have swagger file.




| **Component briefing** | | | | |
|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Validation** | **Comments** |
| Name | Text Box | Yes | • Name cannot be blank<br>• No space allowed at beginning. | Name of operation |
| Http Method | Combo Box One | Yes | | Predefined Values:<br>GET / POST / PUT / PATCH / DELETE |
| Endpoint | Text Box | Yes | • Endpoint cannot be blank<br>• No space allowed at beginning. | Endpoint URL of operation |
| Gateway Prefix | Text Box | | | Gateway Prefix is context path of below formatted URL<br>http://host:port/gateway-prefix/endpoint |
| Service Headers | Collapsible Header & Content | | | Endpoint specific headers<br>Value can either be hardcoded or can be Velocity mapping. |
| Service Query Params | Collapsible Header & Content | | | Endpoint specific query parameters<br>Value can either be hardcoded or can be Velocity mapping. |

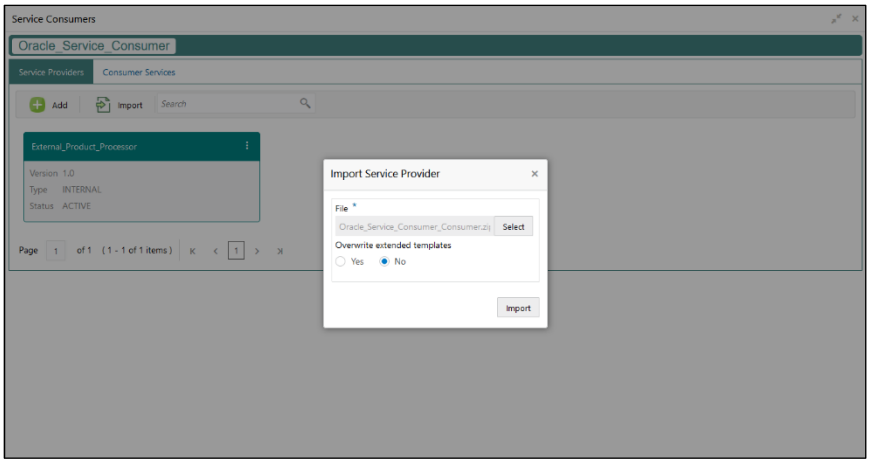
| | | | | |
|---|---|---|---|---|
| Add | Button | | | Adds the endpoint details in the Service list |

In case there is a change in existing endpoint, then same endpoint details need to be entered again with the new changes in order to update the existing provided service information in routing hub.

## 6.2 Import

User can create a service provider by importing the JSON file. User can also import zip file in order to import all the configuration JSON files together (except parent level configuration JSON files).

Navigation: **Service Providers -> Import**




| **Component briefing** | | | | | |
|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Validation** | **Condition** | **Comments** |
| **File** | File picker | Yes | Allows only to select one file | Accepts JSON and ZIP file | Pops up file selection dialog box |
| **Overwrite extended templates** | Radio Button | No | | | Predefined Values: **Yes** / **No**<br><br>**Yes:** This option is for overwriting the extended templates in configuration |

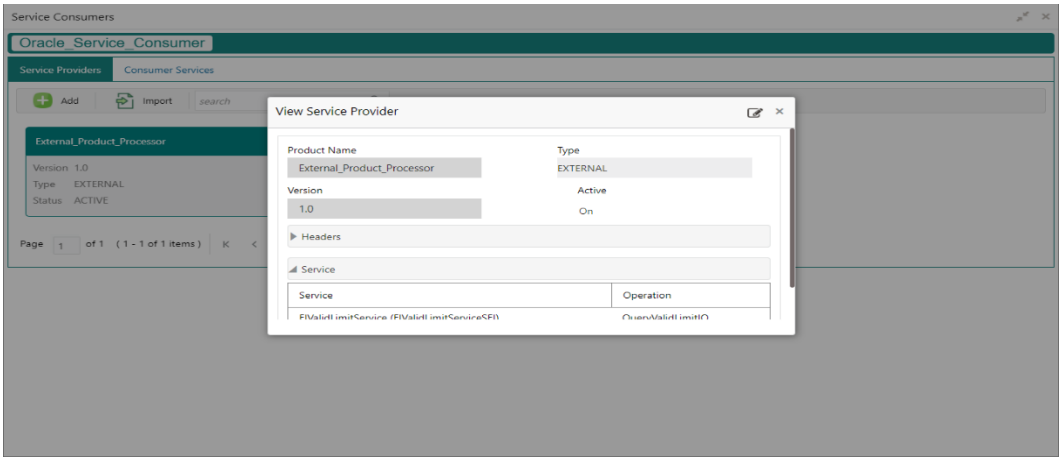
| | | | | | |
|---|---|---|---|---|---|
| | | | | | **No:** This option is for retaining the existing extended templates in configuration.<br>Note: This option is only visible if ZIP file is selected |
| **Import** | Button | | | | Imports Service Provider |

Below data needs to be changed after importing provider configuration file:

- Implementation Host and Port
- Implementation Authentication Password

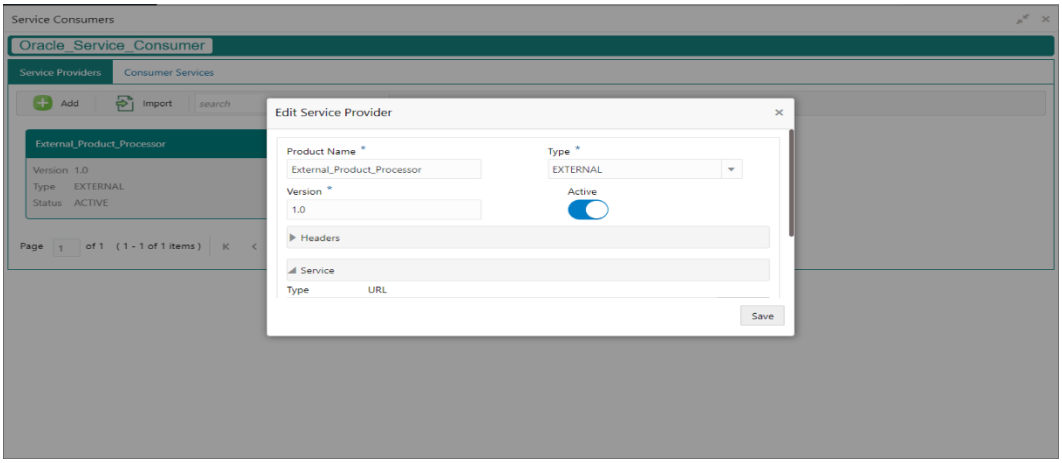
## 6.3 View

User can view provider details and can also switch to edit form by clicking on edit icon.
Navigation: **Service Providers -> Operation Menu (3 dot icon) -> View**



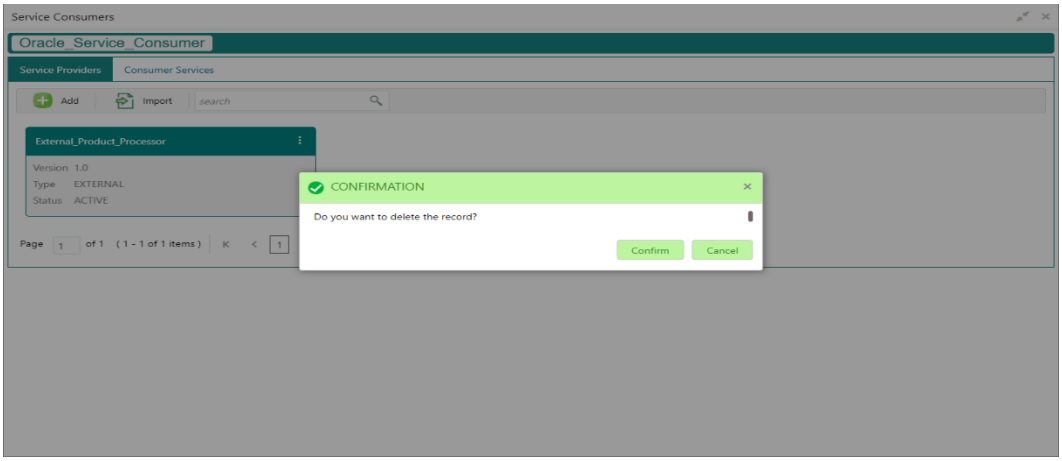

## 6.4 Edit

User can modify the provider details.
Navigation: **Service Providers -> Operation Menu (3 dot icon) -> Edit**



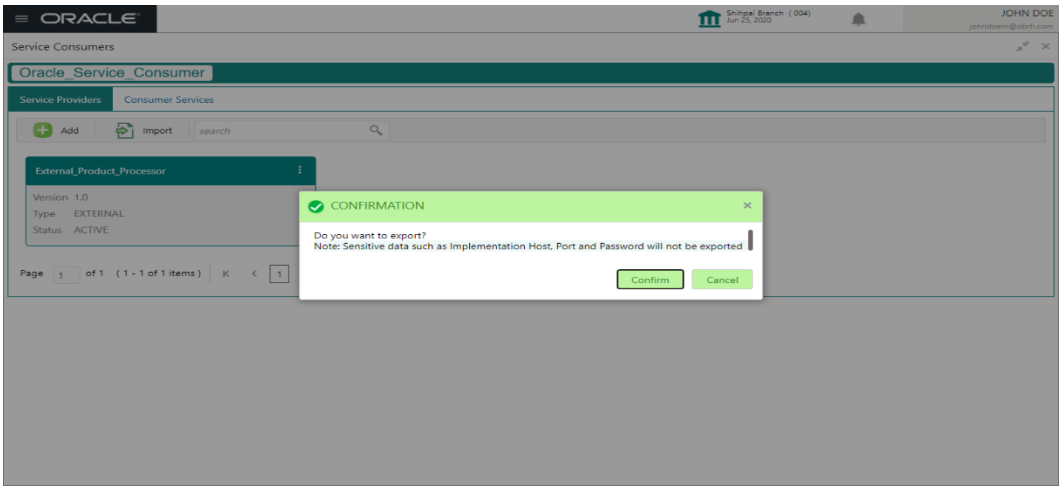

## 6.5 Delete

User can delete the provider.
Navigation: **Service Providers -> Operation Menu (3 dot icon) -> Delete**



## 6.6 Export

User can export the provider configuration as JSON file.
Navigation: **Service Providers -> Operation Menu (3 dot icon) -> Export**



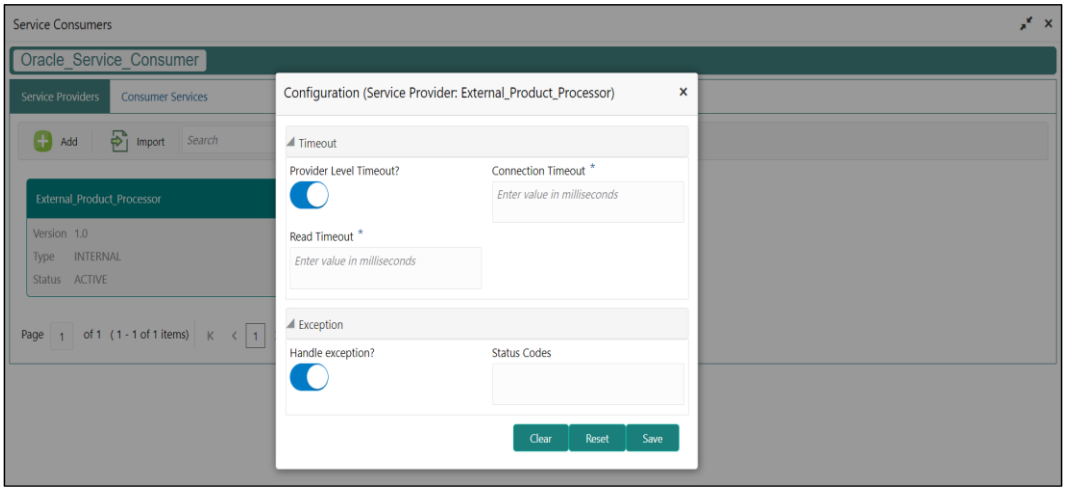

Below data will not be exported:

- 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

## 6.7 Configuration

End-user can configure the properties for failing the routing hub requests.
Navigation: **Service Providers -> Operation Menu (3 dot icon) -> Configuration**



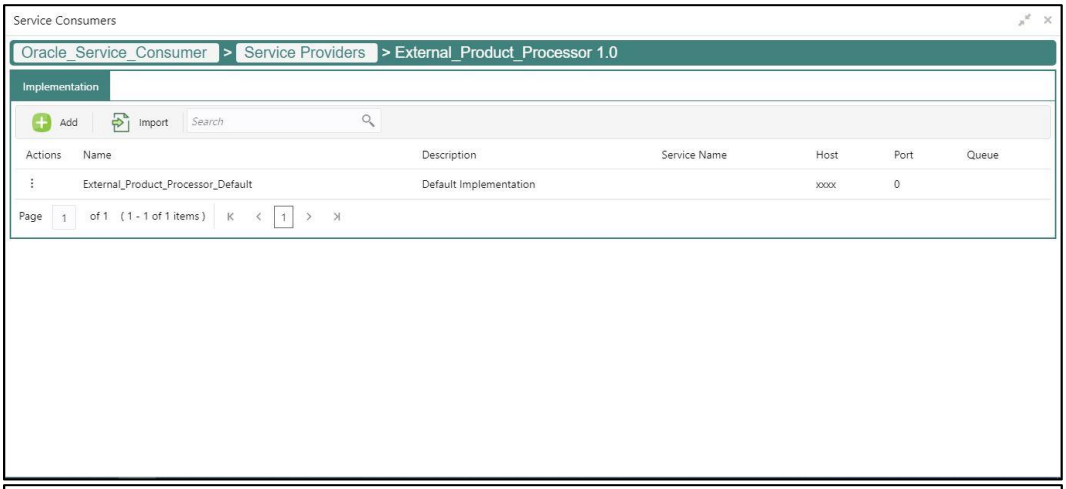
| Component briefing | | | | |
|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Validation** | **Comments** |
| Provider level timeout | Switch | | | This property is used to override the global timeout values.<br>Default value is false. |
| Connection Timeout | Text Box | No | Value should be in milliseconds | This property is used to set the timeout in making the initial connection i.e. connection handshake. |
| Read Timeout | Text Box | No | Value should be in milliseconds | This property is used to set the timeout on waiting to read data. |
| Handle exception | Switch | | | This property is used to fail the routing hub request for failed provider requests.<br>Default value is false. |
| Status Codes | Text Box | No | **Only 4xx and 5xx status codes are allowed as comma-separated values.** | 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. |

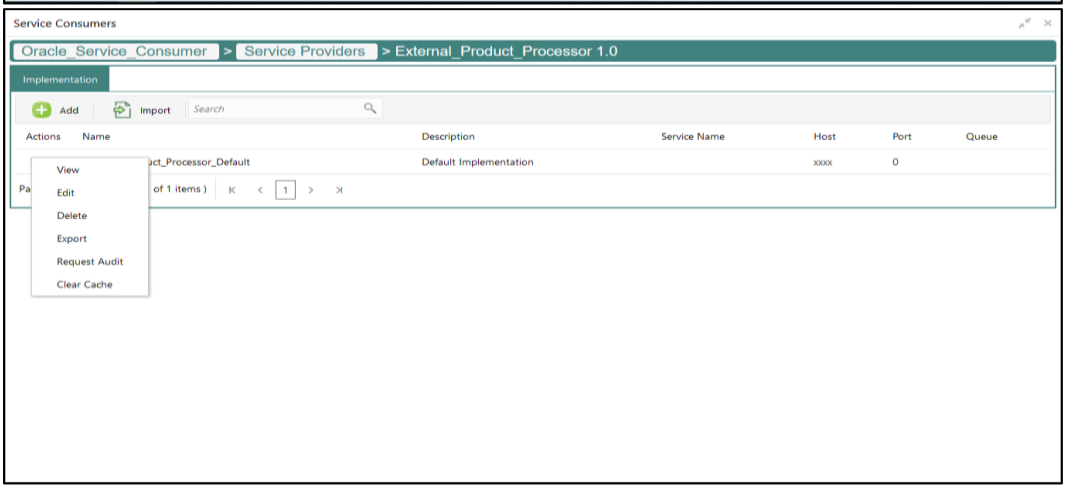
# 7. Implementation

Implementation comprises of Eureka client instance, Queue, Host, Port, authentication and implementation specific service details. Oracle Banking Routing Hub supports webservice and Rest API.

Default implementation is created whenever a new service provider is added.

Navigation: **Core Maintenance -> Routing Hub -> Service Consumers -> <Specific Service Consumer> -> Service Providers -> <Specific Service Provider> -> Implementation**



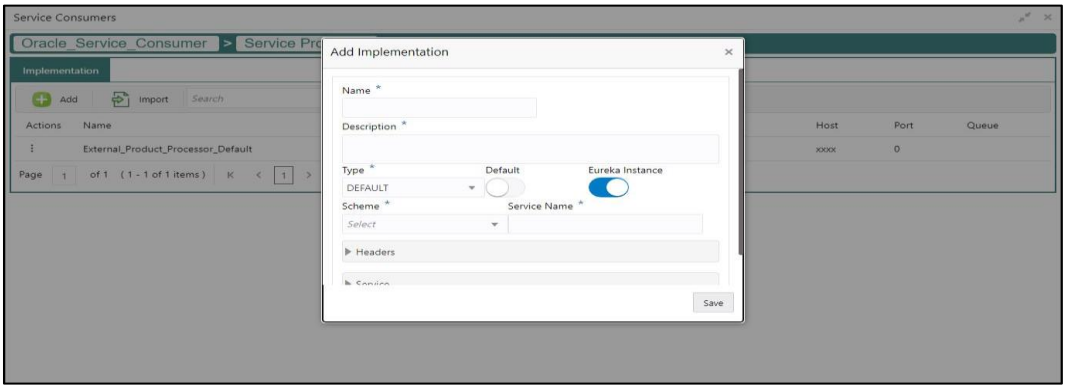




| **Component briefing** | | | |
|---|---|---|---|
| **Component Name** | **Component Type** | **Condition** | **Comments** |
| **<Service Consumer>** | Button | | Navigates back to Service Consumers |

| | | | |
|---|---|---|---|
| **<Service Provider>** | Button | | Navigates back to Service Providers |
| **Add** | Button | | Pops up add dialog |
| **Import** | Button | | Pops up import dialog |
| **Search** | Combo Box One | | Provides search functionality with case insensitive (Implementation Name) |
| **Navigation: Implementation -> 3 dot icon (operation menu)** | | | |
| **View** | menu option | Non-editable | Pops up view dialog |
| **Edit** | menu option | | Pops up edit dialog |
| **Delete** | menu option | | |
| **Export** | menu option | | Exports in JSON |
| **Configuration** | menu option | | Pops up configuration dialog |
| **Request Audit** | menu option | | Pops up request audit log |
| **Clear Cache** | Menu option | | Clears SOAP client cache |

## 7.1 Add

User can create Implementation manually.
Navigation: **Implementation-> Add**



| Component briefing | | | | | |
|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Validation** | **Comments** |
| **Name** | Text Box | Yes | Alphanumeric with special characters | • Name cannot be blank<br>• Enter 0 or more characters, up to a maximum of 255.<br>• No numeric value at beginning and no space allowed. | Unique implementation name |
| **Description** | Text Area | Yes | Alphanumeric with special characters | • Description cannot be blank<br>• Enter 0 or more characters, up to a maximum of 1000.<br>• No space allowed at beginning or ending of the value. | |
| **Type** | Combo Box One | Yes | | | Predefined Values:<br>DEFAULT / QUEUE<br>**Note:** DEFAULT type is for REST and SOAP API calls. |
| **Default** | Switch | | | | Each type can have one default implementation. |

| | | | | | |
|---|---|---|---|---|---|
| **Eureka Instance** | Switch | | | | **Eureka Instance** option is available only for internal providers and default type.<br>By default, **Eureka Instance** will be toggled ON for internal providers and OFF for external providers. |
| **Scheme** | Combo Box One | Yes | | | Scheme option is available only for default type.<br>Predefined Values:<br>HTTPS / HTTP |
| **Service Name** | Text Box | Yes | | • Service Name cannot be blank<br>• Enter 0 or more characters, up to a maximum of 255.<br>No space allowed. | If **Eureka Instance** is toggled ON and type is default, then only service name is required. |
| **Host** | Text Box | Yes | Alphanumeric with special characters | • Host cannot be blank<br>• Enter 0 or more characters, up to a maximum of 255.<br>• No space allowed. | If **Eureka Instance** is toggled OFF and type is default, then only host and port is required. |
| **Port** | Text Box | No | Number | • Enter 0 or more characters, up to a maximum of 6. | If **Eureka Instance** is toggled off and type is default, then only host and port is required. |

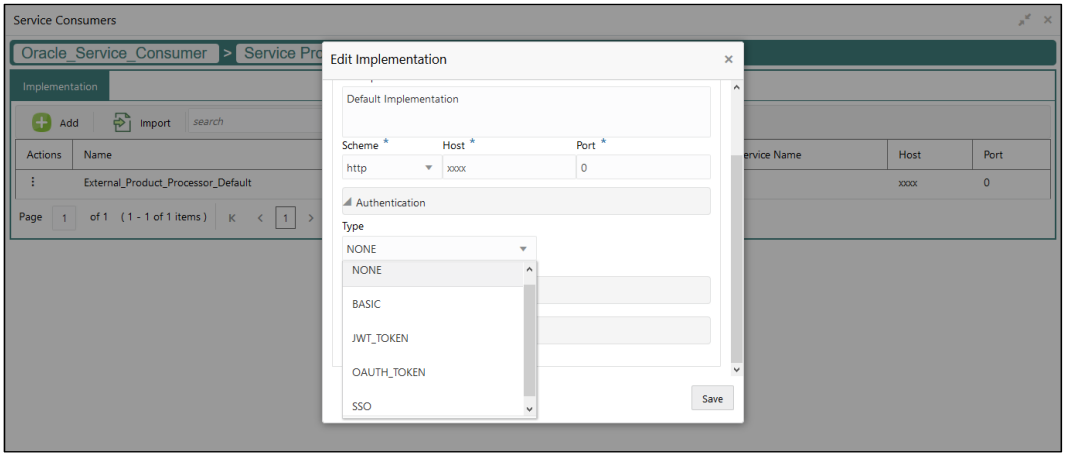
| | | | | | |
|---|---|---|---|---|---|
| | | | | • Only numeric value allowed. | |
| **Authentication** | Collapsible Header & Content | | | | Authentication option is available only when **Eureka Instance** is toggled OFF and type is default. |
| **Headers** | Collapsible Header & Content | | | | Header option is available only when type is default. |
| **Service** | Collapsible Header & Content | | | | Service option is available only when type is default. |
| **Queue** | Content | | | | Queue content is available only when type is queue. |
| **Save** | Button | | | | Saves the Implementation |

## 7.1.1 Authentication

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

In case of no authentication, NONE needs to be set as Authentication Type.
In case of identity propagation, SSO needs to be set as Authentication Type.
In case of OIC integration, OAUTH_TOKEN_OIC needs to be set as Authentication Type.
In case of JWT, OAUTH_TOKEN and OAUT_TOKEN_OIC, token will be cached by default.

Process of configuration of these is described below.

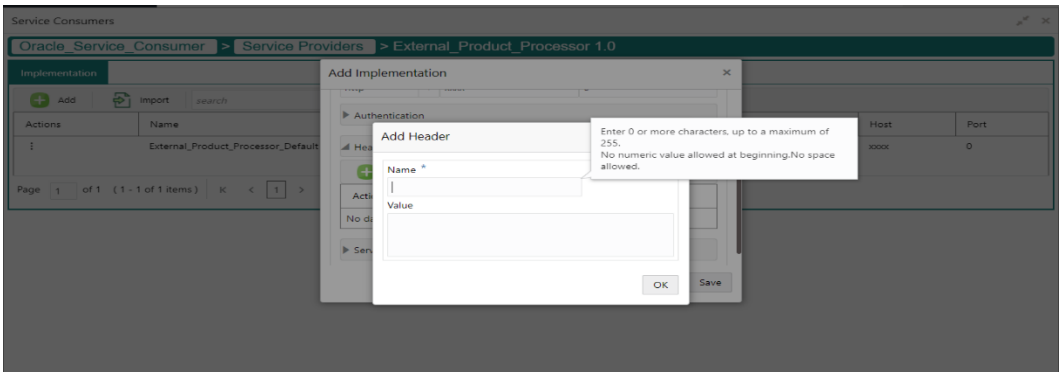


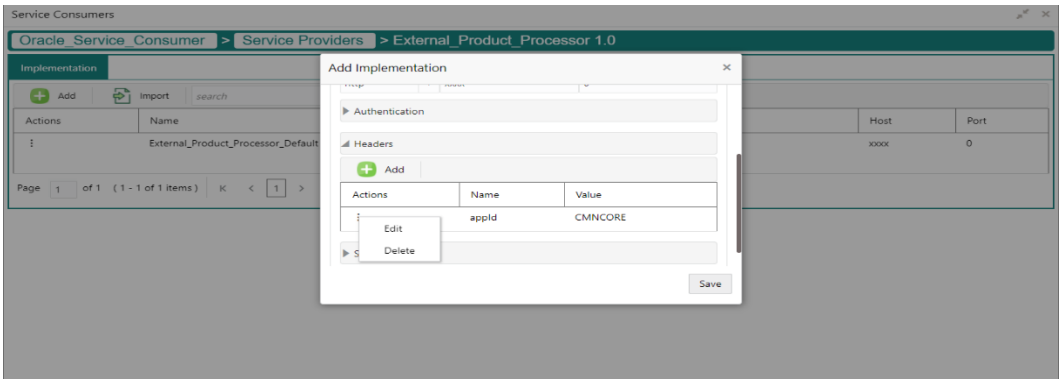

| Component briefing | | | | | |
|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Validation** | **Comments** |
| **Type** | Combo Box One | Yes | | | Predefined Values: NONE / BASIC / JWT_TOKEN / OAUTH_TOKEN / SSO / OAUTH_TOKEN_OIC |
| **Username** | Text Box | | Alphanumeric with special characters | • Username cannot be blank<br>• Enter 0 or more characters, up to a maximum of 255. | Username is mandatory if authentication type is BASIC / JWT_TOKEN / OAUTH_TOKEN |

| | | | | | |
|---|---|---|---|---|---|
| | | | | • No numeric value at beginning and no space allowed. | |
| **Password** | Text Box | | Alphanumeric with special characters | Password cannot be blank | Password is mandatory if authentication type is BASIC / JWT_TOKEN / OAUTH_TOKEN |

## 7.1.2 Headers

External product processor might require some standard headers to be passed along with the request.
User can specify the headers which are required by service endpoints for specific implementation but not present in swagger file.
These headers can be configured in Oracle Banking Routing Hub using the steps given below

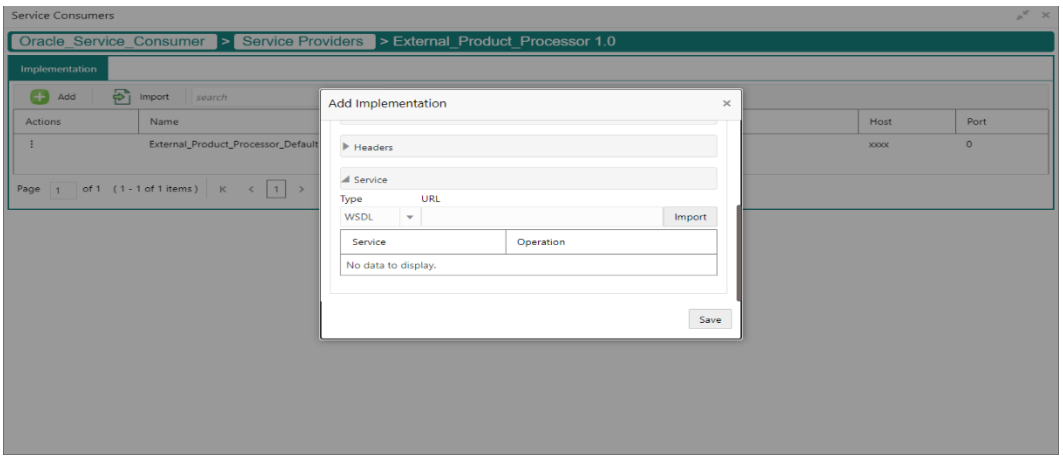



| **Component briefing** | | | | | |
|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Validation** | **Comments** |

| | | | | | |
|---|---|---|---|---|---|
| **Add** | Button | | | | Pops up add dialog |
| **Navigation: Implementation -> Headers -> 3 dot icon (operation menu)** | | | | | |
| **Edit** | menu option | | | | Pops up edit dialog |
| **Delete** | menu option | | | | Deletes header |
| **Navigation: Implementation -> Headers -> Add** | | | | | |
| Name | Text Box | Yes | Alphanumeric with special characters | • Name cannot be blank<br>• Enter 0 or more characters, up to a maximum of 255.<br>• No numeric value at beginning and no space allowed. | |
| Value | Text Area | Yes | Alphanumeric with special characters | • Enter 0 or more characters, up to a maximum of 255. | Value can either be hardcoded or can be Velocity mapping. |
| OK | Button | | | | Saves the header details and displays it in the list |

## 7.1.3 Service



| Component briefing | | | |
|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Comments** |
| Type | Combo Box One | Yes | Predefined Values: WSDL / SWAGGER / OTHERS |
| URL | Text Box | Yes | Service URL of the file location |
| Gateway Prefix | Text Box | | Gateway Prefix is context path of below formatted URL<br>http://host:port/gateway-prefix/endpoint |
| Import | Button | | Extracts the service information from URL and displays it in the Service list |

### *7.1.3.1 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.

In case there is a change in wsdl file, then same wsdl file need to be imported again in order to update the provided service information in routing hub.

### *7.1.3.2 SWAGGER*

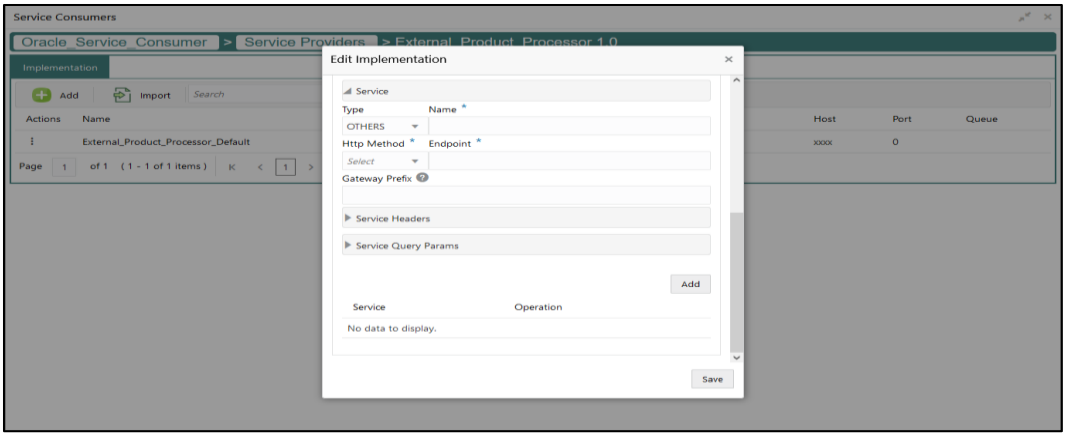
Swagger is an Interface Description Language for describing RESTful APIs expressed using JSON.

Currently, Swagger 2.0 & OpenAPI 3.0 both are supported.

In case 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.

### *7.1.3.3 OTHERS*

"OTHERS" option is for adding REST API details manually when provider does not have swagger file.

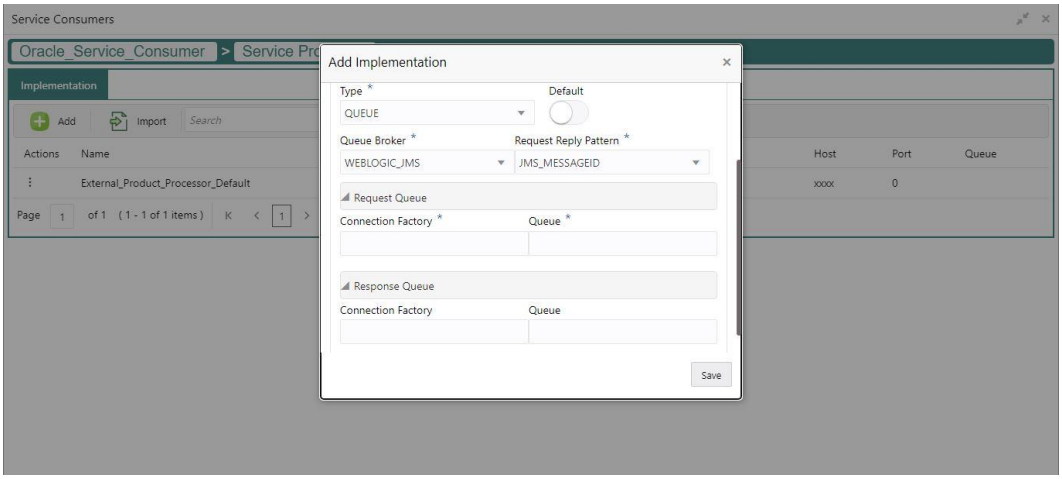



| **Component briefing** | | | | |
|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Validation** | **Comments** |
| Name | Text Box | Yes | • Name cannot be blank<br>• No space allowed at beginning. | Name of operation |
| Http Method | Combo Box One | Yes | | Predefined Values:<br>GET / POST / PUT / PATCH / DELETE |
| Endpoint | Text Box | Yes | • Endpoint cannot be blank<br>• No space allowed at beginning. | Endpoint URL of operation |
| Gateway Prefix | Text Box | | | Gateway Prefix is context path of below formatted URL<br>http://host:port/gateway-prefix/endpoint |
| Service Headers | Collapsible Header & Content | | | Endpoint specific headers<br>Value can either be hardcoded or can be Velocity mapping. |
| Service Query Params | Collapsible Header & Content | | | Endpoint specific query parameters<br>Value can either be hardcoded or can be Velocity mapping. |

| | | | | |
|---|---|---|---|---|
| Add | Button | | | Adds the endpoint details in the Service list |

In case there is a change in existing endpoint, then same endpoint details need to be entered again with the new changes in order to update the existing provided service information in routing hub.

### 7.1.4 Queue

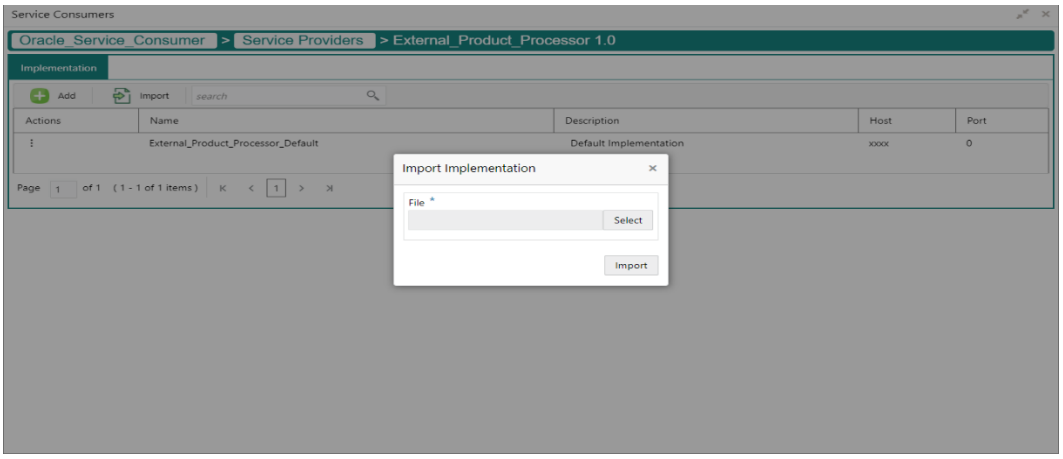



| **Component briefing** | | | |
|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Comments** |
| Queue Broker | Combo Box One | Yes | Predefined Values:<br>WEBLOGIC_JMS |
| Request Reply Pattern | Combo Box One | Yes | Predefined Values:<br>JMS_MESSAGEID / JMS_CORRELATIONID<br>JMS_MESSAGEID is default request-reply pattern. |
| Connection Factory | Text Box | | Connection Factory is JNDI based connection factory name which is used to create connection for JMS client.<br>Request Connection Factory is mandatory, and Response Connection Factory is optional. |

| | | | |
|---|---|---|---|
| | | | Response Connection Factory is needed when destination is going to respond back after processing the request. |
| Queue | Text Box | | Queue Name is JNDI based destination name.<br>Request Queue Name is mandatory, and Response Queue Name is optional.<br>Response Queue Name is needed when destination is going to respond back after processing the request. |

## 7.2 Import

User can create an implementation by importing the JSON file. User can also import zip file in order to import all the configuration JSON files together (except parent level configuration JSON files).
Navigation: **Implementation -> Import**




| **Component briefing** | | | | | | |
|---|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Validation** | **Condition** | **Comments** |
| **File** | File picker | Yes | | Allows only to select one file | Accepts JSON and ZIP file | Pops up file selection dialog box |

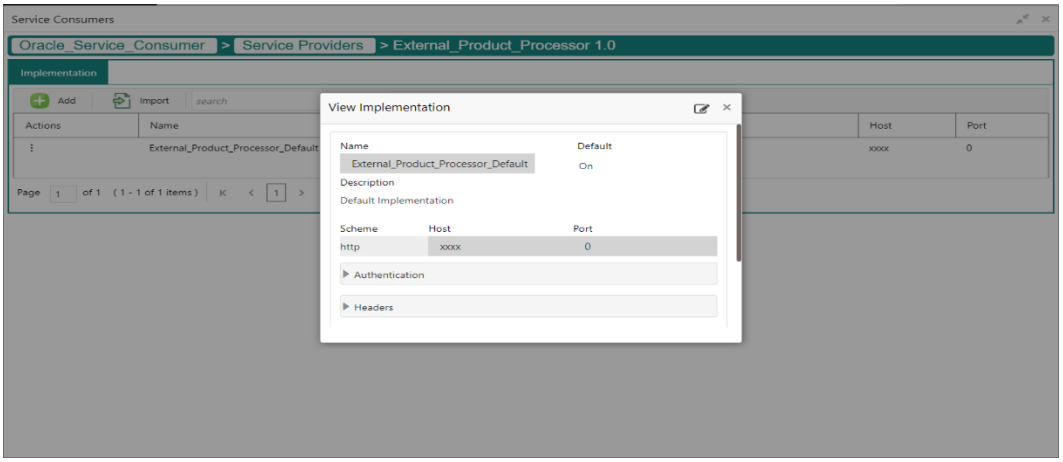
| | | | | | | |
|---|---|---|---|---|---|---|
| **Import** | Button | | | | | Imports Implementation |

Below data needs to be changed after importing implementation configuration file:

- Implementation Host and Port
- Implementation Authentication Password

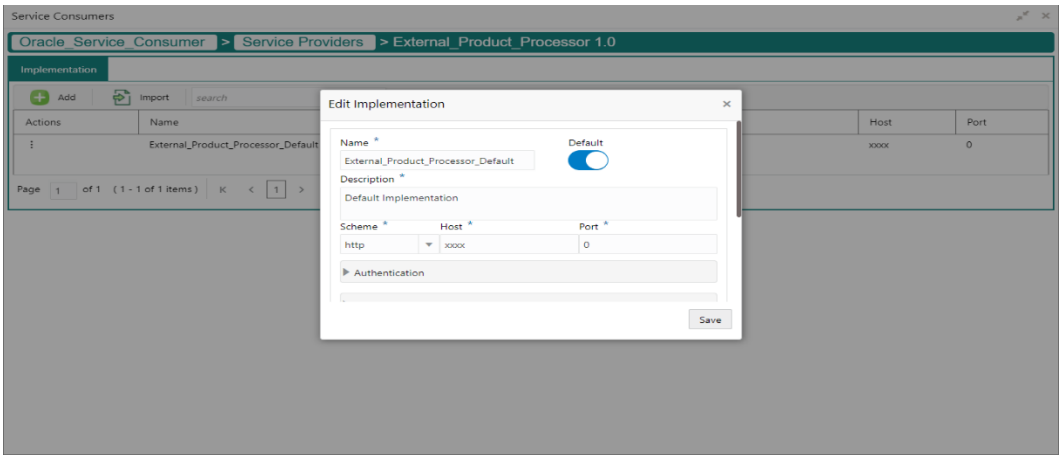
## 7.3 View

User can view implementation details and can also switch to edit form by clicking on edit icon.
Navigation: **Implementation -> Operation Menu (3 dot icon) -> View**



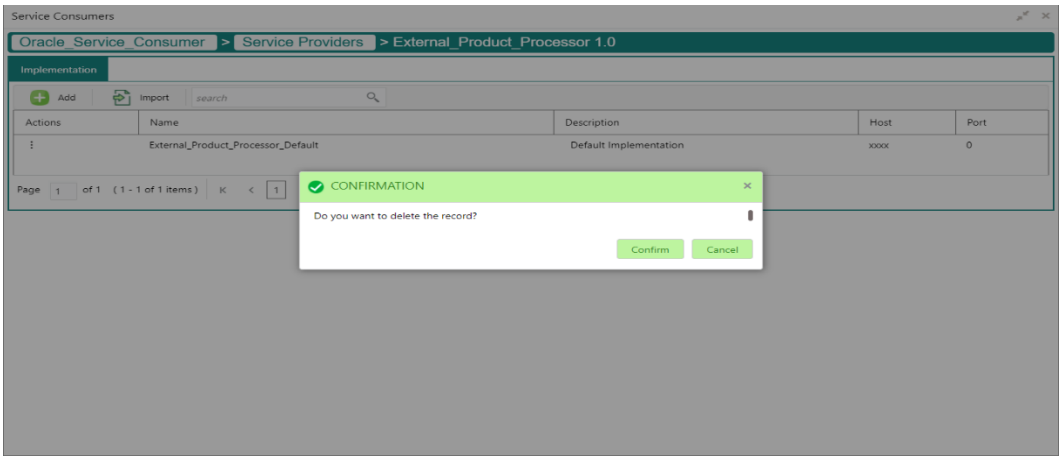

## 7.4 Edit

User can modify the implementation details.
Navigation: **Implementation -> Operation Menu (3 dot icon) -> Edit**



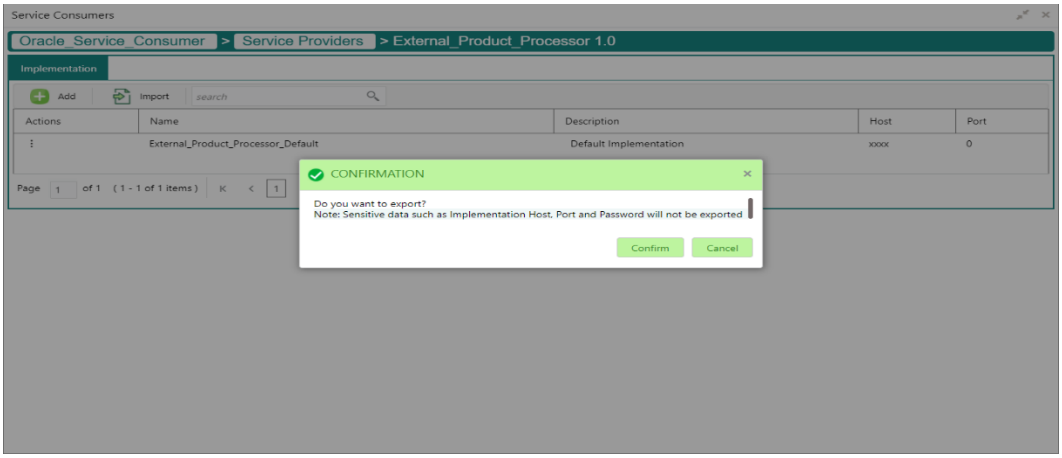
## 7.5 Delete

User can delete the implementation.
Navigation: **Implementation -> Operation Menu (3 dot icon) -> Delete**



## 7.6 Export

User can export the implementation configuration as JSON file.
Navigation: **Implementation -> Operation Menu (3 dot icon) -> Export**



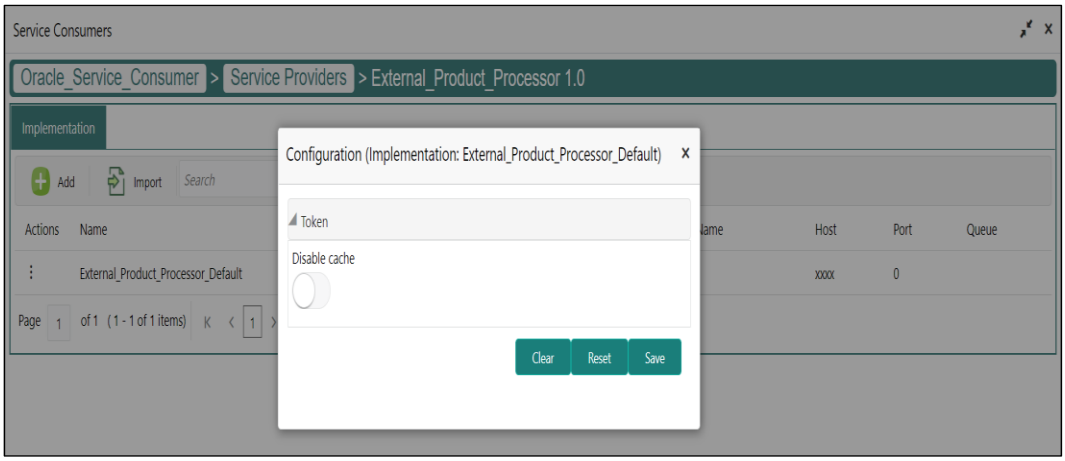

**Below data will not be exported:**

- 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

## 7.7 Configuration

End-user can configure the property to disable the token caching.
Navigation: **Implementation -> Operation Menu (3 dot icon) -> Configuration**

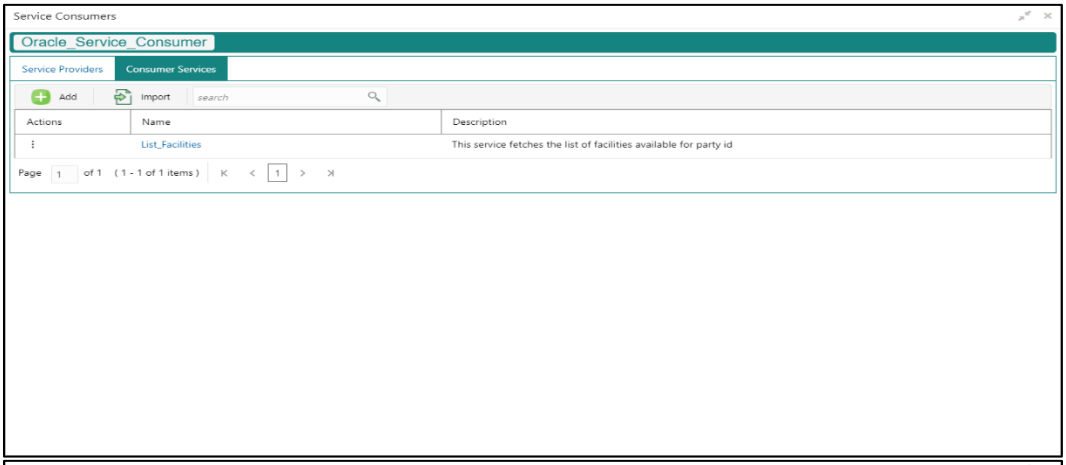


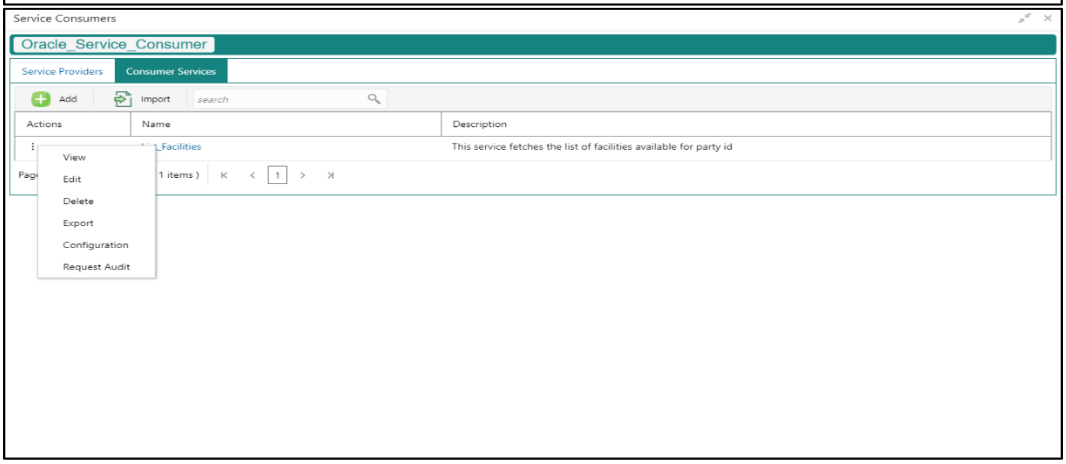
| Component briefing | | | | |
|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Validation** | **Comments** |
| Disable cache | Switch | | | This property is used to disable the token caching.<br>Default value is false. |

# 8. Consumer Services

Consumer Services defines the service ID which is going to be send from service consumer. This will also cater transformation and route definition as well.

Consumer Services comprises of source integration details.
Navigation: **Core Maintenance -> Routing Hub -> Service Consumers -> <Specific Service Consumer> -> Consumer Services**



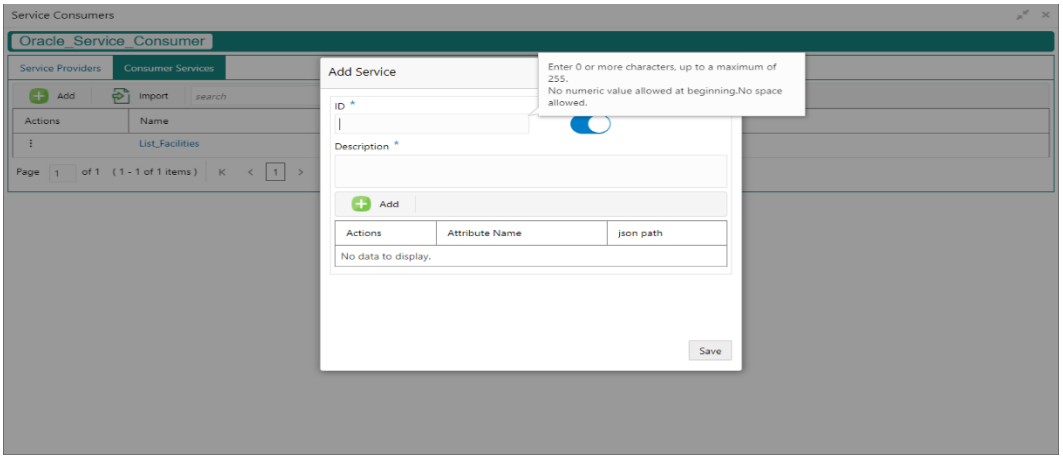

| **Component briefing** | | | |
|---|---|---|---|
| **Component Name** | **Component Type** | **Condition** | **Comments** |
| **<Service Consumer>** | Button | | Navigates back to Service Consumers |
| **Add** | Button | | Pops up add dialog |

| | | | |
|---|---|---|---|
| **Import** | Button | | Pops up import dialog |
| **Search** | Combo Box One | | Provides search functionality with case insensitive (Consumer Service Name) |
| **Navigation: Consumer Services -> 3 dot icon (operation menu)** | | | |
| **View** | menu option | Non-editable | Pops up view dialog |
| **Edit** | menu option | | Pops up edit dialog |
| **Delete** | menu option | | |
| **Export** | menu option | | Exports in JSON |
| **Configuration** | menu option | | Pops up configuration dialog |
| **Request Audit** | menu option | | Pops up request audit log |

## 8.1 Add

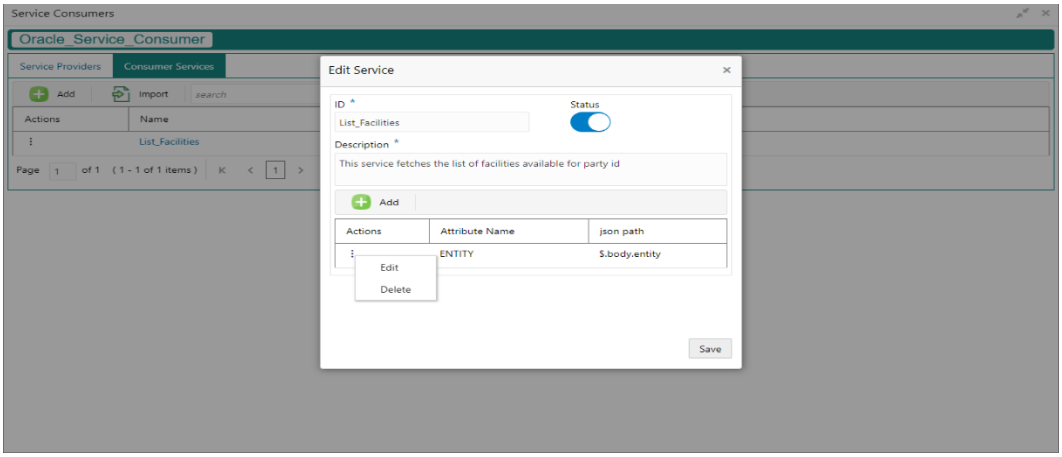
User can create Consumer Service manually.

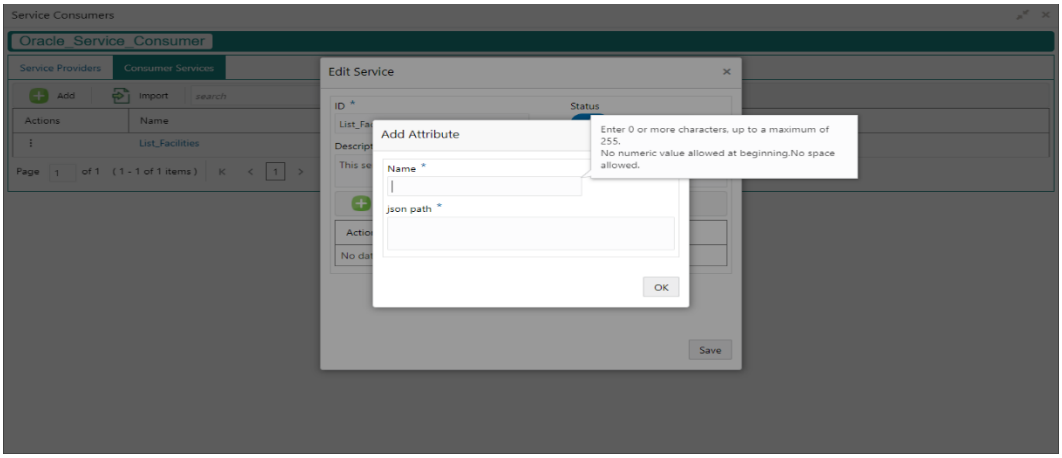
Navigation: Consumer Services-> Add



| Component briefing | | | | | |
|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Validation** | **Comments** |
| Id | Text Box | Yes | Alphanumeric with special characters | Name cannot be blank<br><br>Enter 0 or more characters, up to a maximum of 255.<br><br>No numeric value at beginning and no space allowed. | Unique Service Id |
| Status | Switch | Yes | | | Predefined Values: ACTIVE / INACTIVE<br>If consumer service is marked as inactive, then all related routes will be stopped. |
| Description | Text Area | Yes | Alphanumeric with special characters | Description cannot be blank<br><br>Enter 0 or more characters, up to a maximum of 1000.<br><br>No space allowed at beginning or ending of the value. | |
| Attributes | Table Content | | | | |
| Save | Button | | | | Saves the Consumer Service |

## 8.1.1 **Attributes**



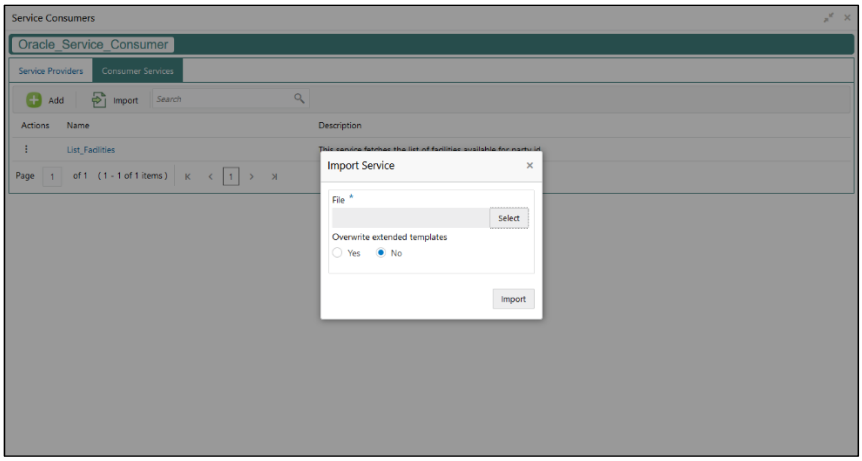

| **Component briefing** | | | | | | |
|---|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Validation** | **Condition** | **Comments** |
| | | | | | | |
| **Add** | Button | | | | | Pops up add dialog |
| **Navigation: Consumer Services -> Headers -> 3 dot icon (operation menu)** | | | | | | |
| **Edit** | menu option | | | | | Pops up edit dialog |
| **Delete** | menu option | | | | | Deletes attribute |

| Navigation: Consumer Services -> Headers -> Add | | | | | | |
|---|---|---|---|---|---|---|
| Name | Text Box | Yes | Alphanumeric with special characters | Name cannot be blank<br><br>Enter 0 or more characters, up to a maximum of 255.<br><br>No numeric value at beginning and no space allowed. | | |
| JSON Path | Text Area | Yes | Alphanumeric with special characters | Value cannot be blank<br><br>Enter 0 or more characters, up to a maximum of 255.<br><br>No space allowed. | | Value will be JSON path. |
| OK | Button | | | | | Saves the header details and displays it in the list |

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

## 8.2 Import

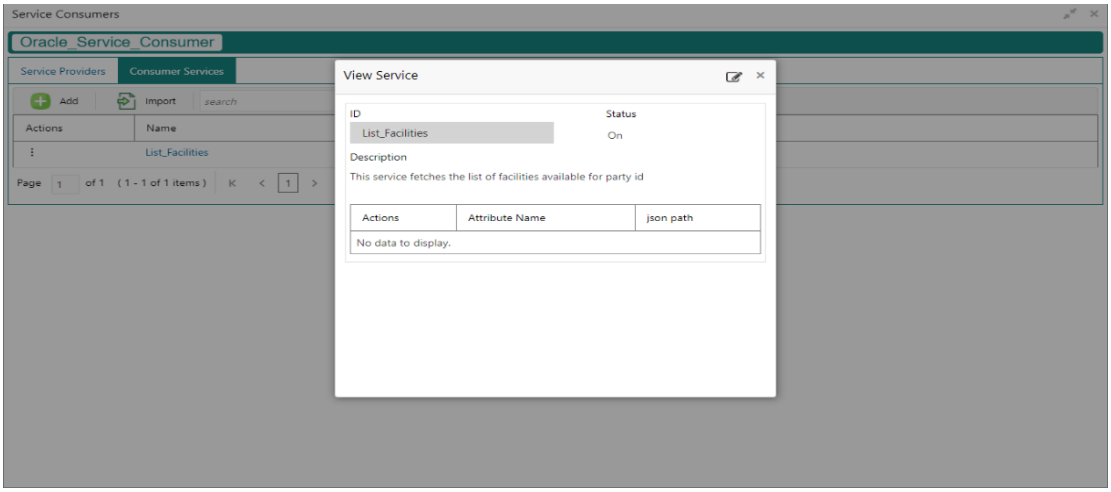
User can create a consumer service by importing the JSON file.
User can also import zip file in order to import all the configuration JSON files together (except parent level configuration JSON files).
Navigation: **Consumer Services -> Import**




| Component briefing | | | | | | |
|---|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Validation** | **Condition** | **Comments** |
| **File** | File picker | Yes | | Allows only to select one file | Accepts JSON and ZIP file | Pops up file selection dialog box |
| **Overwrite extended templates** | Radio Button | Yes | | | | Predefined Values:<br>Yes / No<br>Yes: This option is for overwriting the extended templates in configuration and<br>No: This option is for retaining the existing extended templates in configuration. |
| **Import** | Button | | | | | Imports Consumer Service |

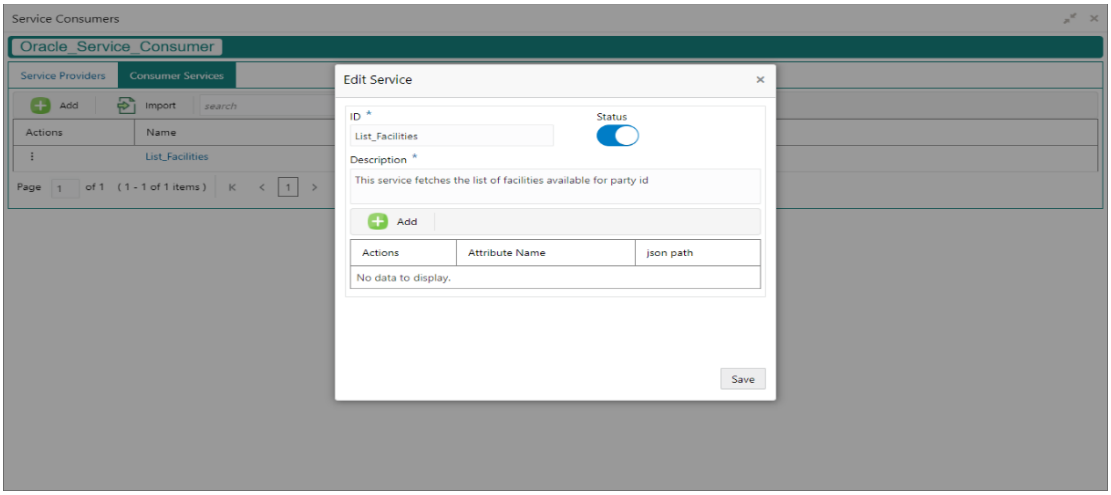
## 8.3 View

User can view consumer service details and can also switch to edit form by clicking on edit icon.
Navigation: **Consumer Services -> Operation Menu (3 dot icon) -> View**



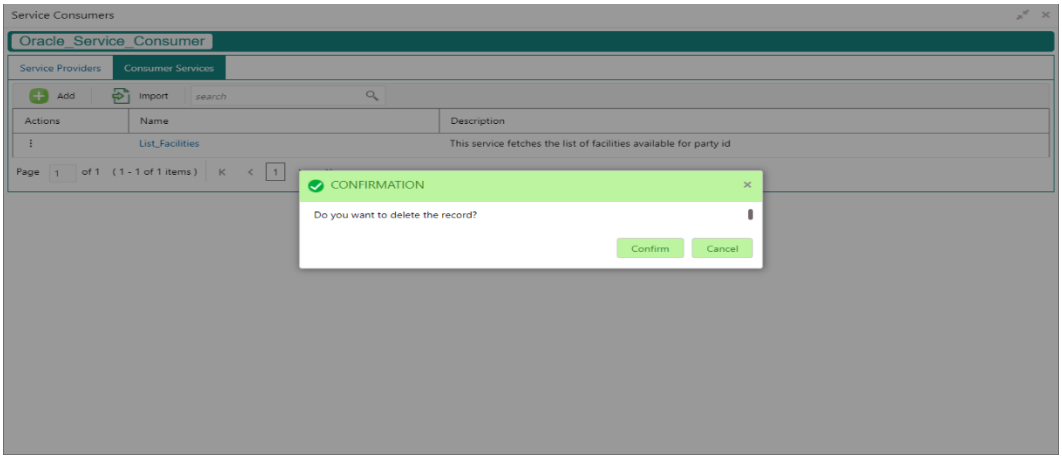

## 8.4 Edit

User can modify the consumer service details.
Navigation: **Consumer Services -> Operation Menu (3 dot icon) -> Edit**



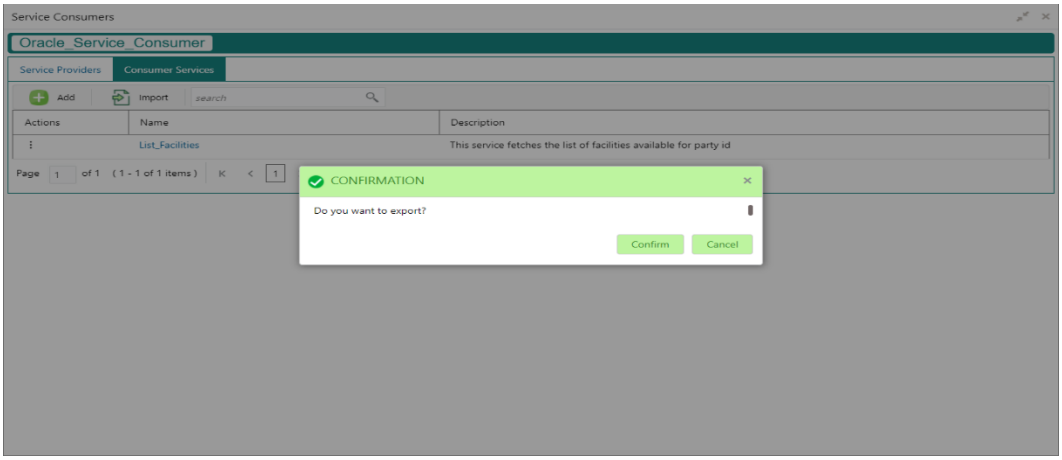

## 8.5 Delete

User can delete the consumer service.
Navigation: **Consumer Services -> Operation Menu (3 dot icon) -> Delete**




## 8.6 Export

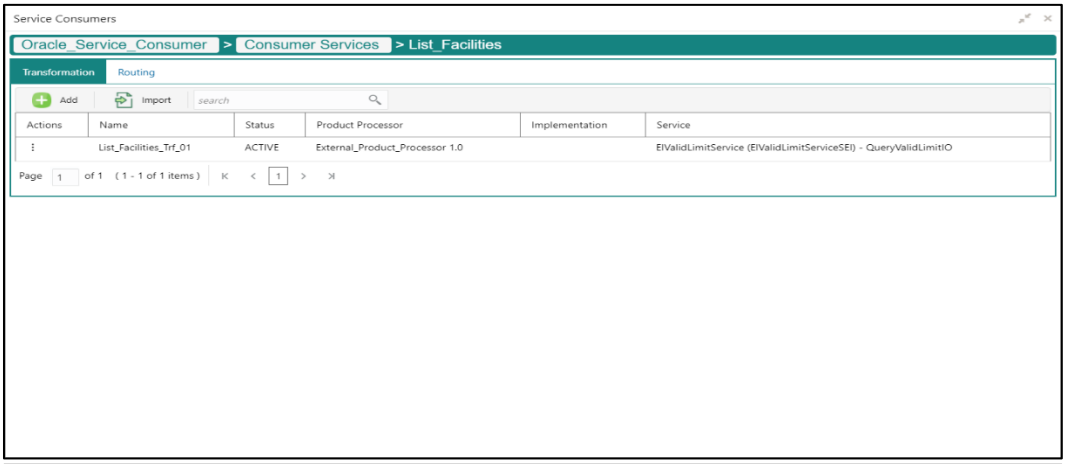
User can export the consumer service configuration as JSON file.
Navigation: **Consumer Services -> Operation Menu (3 dot icon) -> Export**

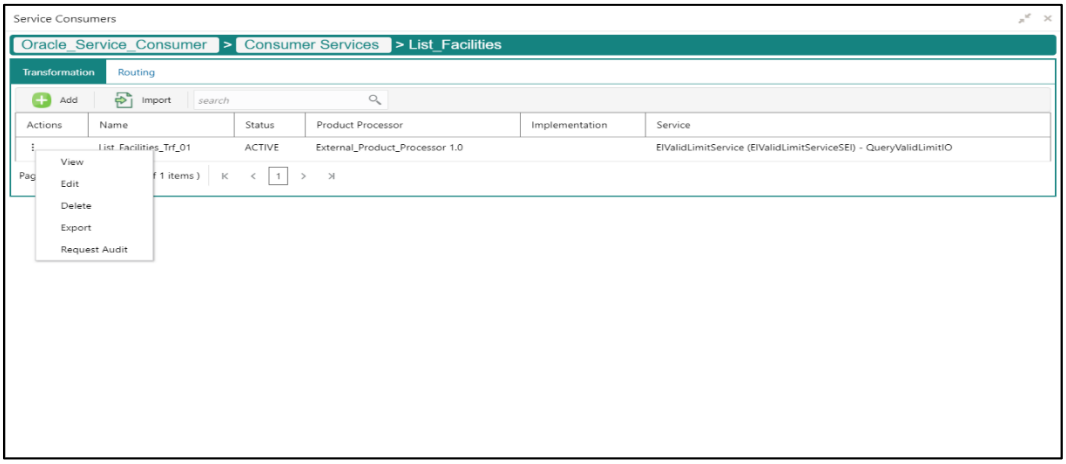


# 9. Transformation

Transformation acts as an assembling and transforming data from source to destination and vice-versa. This will take place under consumer service. This converts data of service consumer into service provider

**Navigation: Core Maintenance -> Routing Hub -> Service Consumers -> <Specific Service Consumer> -> Consumer Services -> <Specific Consumer Service> -> Transformation**



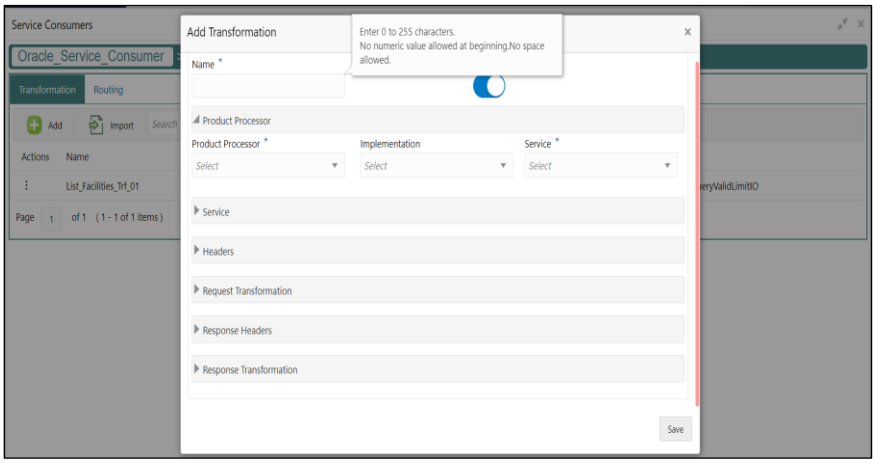




| **Component briefing** | | | |
|---|---|---|---|
| **Component Name** | **Component Type** | **Condition** | **Comments** |
| **<Service Consumer>** | Button | | Navigates back to Service Consumers |
| **<Consumer Service>** | Button | | Navigates back to Consumer Services |

| | | | |
|---|---|---|---|
| **Add** | Button | | Pops up add dialog |
| **Import** | Button | | Pops up import dialog |
| **Search** | Combo Box One | | Provides search functionality with case insensitive (Transformation Name) |
| **Navigation: Transformation -> 3 dot icon (operation menu)** | | | |
| **View** | menu option | Non-editable | Pops up view dialog |
| **Edit** | menu option | | Pops up edit dialog |
| **Delete** | menu option | | |
| **Export** | menu option | | Exports in JSON |
| **Request Audit** | menu option | | Pops up request audit log |

## 9.1 Add

User can create Transformation manually.
Navigation: **Transformation-> Add**




| **Component briefing** | | | | | |
|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Validation** | **Comments** |

| | | | | | |
|---|---|---|---|---|---|
| **Name** | Text Box | Yes | Alphanumeric with special characters | • Name cannot be blank<br>• Enter 0 or more characters, up to a maximum of 255.<br>• No numeric value at beginning and no space allowed. | Unique Transformation Name |
| **Active** | Switch | | | | Predefined Values: ACTIVE/INACTIVE<br>If transformation is marked as inactive, then user will not be able to select transformation in routing. |
| **Product Processor** | Collapsible Header | | | | |
| **Product Processor** | Combo Box One | Yes | | | Displays provider list relevant to consumer |
| **Implementation** | Combo Box One | | | | Displays implementation list relevant to selected provider |
| **Service** | Combo Box One | Yes | | | Displays service list relevant to selected provider and implementation |
| **Service** | Collapsible Header & Content | | | | Displays service details of selected service |
| **Headers** | Collapsible Header & Content | | | | Displays header list relevant to selected provider, implementation and service<br>User can change the header values.<br>Value can either be hardcoded or can be Velocity mapping. |

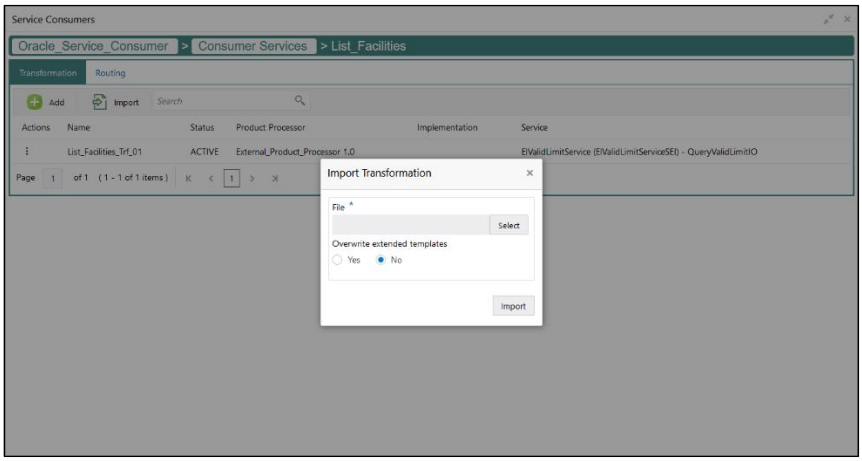
| | | | | | |
|---|---|---|---|---|---|
| **Path Params** | Collapsible Header & Content | | | | Displays path param list relevant to selected service<br>User can change the param values.<br>Value can either be hardcoded or can be Velocity mapping. |
| **Query Params** | Collapsible Header & Content | | | | Displays query param list relevant to selected service<br>User can change the param values.<br>Value can either be hardcoded or can be Velocity mapping. |
| **Request Transformation** | Collapsible Header | | | | |
| **Body Type** | Combo Box One | | | | Predefined Values: RAW / FORM_DATA<br>Note: This option is only visible if selected service is REST service. |
| **Type** | Combo Box One | | | | Predefined Values: VELOCITY / JSLT / XSLT |
| **Template** | Text Area | | | | User has to define the kernel template in which provider accepts.<br>Refer Transformation Type for syntax and Multipart Request for sample request if body type is FORM_DATA |
| **Extended Template** | Text Area | | | | User has to define the custom template in order to extend the kernel template.<br>Please refer Extensibility and Transformation Type for syntax |

| | | | | | |
|---|---|---|---|---|---|
| | | | | | Note: This option is not visible if body type is FORM_DATA. |
| **Response Headers** | Collapsible Header & Content | | | | Response Headers is used for specifying additional headers required to be part of routing hub response headers.<br>Value can either be hardcoded or can be Velocity mapping. |
| **Type** | Combo Box One | | | | Predefined Values: VELOCITY / JSLT / XSLT |
| **Template** | Text Area | | | | User has to define the kernel template in which consumer accepts.<br>Refer Transformation Type for syntax |
| **Mocking required?** | Switch | | | | Predefined Values:<br>YES / NO<br>If flag is marked as yes, then routing hub will return the mocked template output (with extended template output if mentioned) to consumer without invoking provider API. |
| **Mock Template** | Text Area | | | | User has to define the kernel mocked template in which consumer accepts.<br>Refer Transformation Type for syntax |
| **Extended Template** | Text Area | | | | User has to define the custom template in order to extend the kernel template.<br>Refer Extensibility and Transformation Type for syntax |

| | | | | | |
|---|---|---|---|---|---|
| **Save** | Button | | | | Saves transformation details |

## 9.2 Import

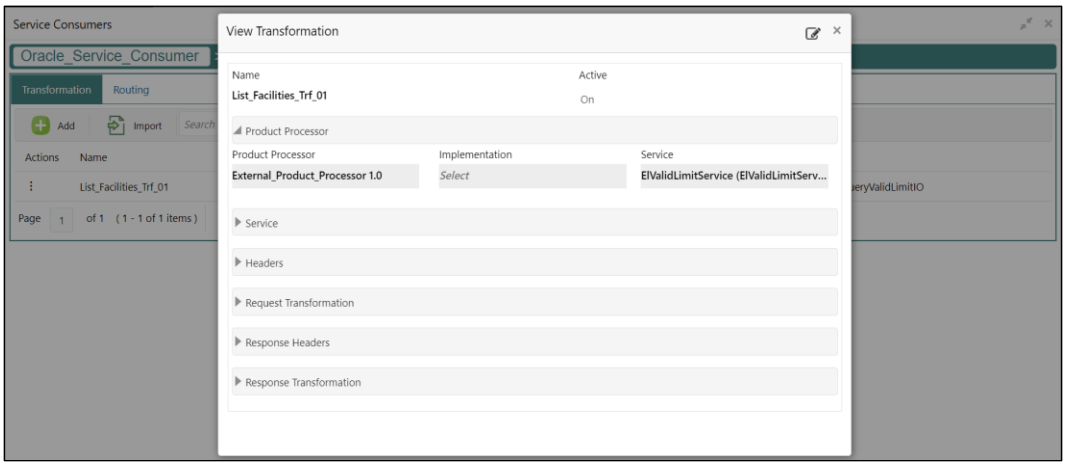
User can create a transformation by importing the JSON file.
User can also import zip file in order to import all the configuration JSON files together (except parent level configuration JSON files).
Navigation: **Transformation -> Import**




| **Component briefing** | | | | | |
|---|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Validation** | **Condition** | **Comments** |
| **File** | File picker | Yes | • Allows only to select one file | Accepts JSON and ZIP file | Pops up file selection dialog box |
| **Overwrite extended templates** | Radio Button | Yes | | | Predefined Values:<br>Yes / No<br><br>Yes: This option is for overwriting the extended templates in configuration<br>No: This option is for retaining the existing extended templates in configuration. |
| **Import** | Button | | | | Imports Transformation |

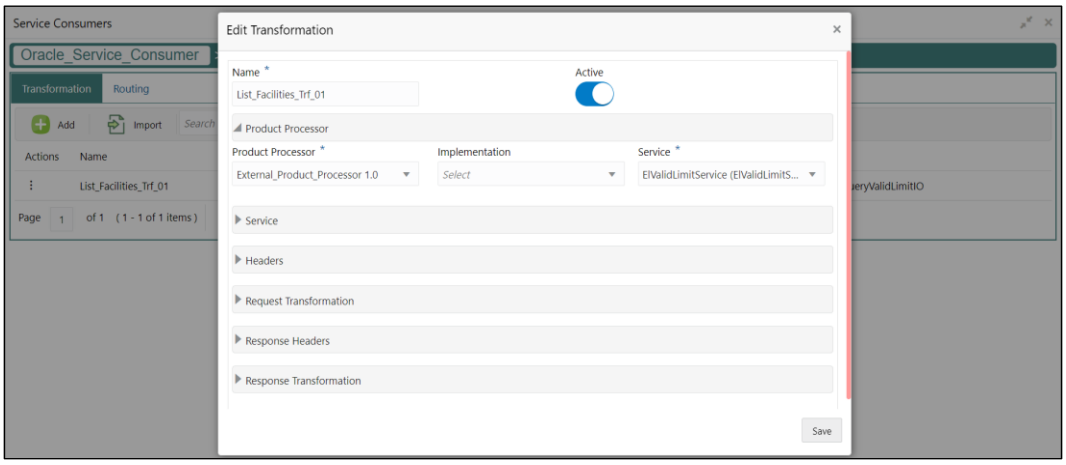
## 9.3 View

User can view transformation details and can also switch to edit form by clicking on edit icon.
Navigation: **Transformation -> Operation Menu (3 dot icon) -> View**



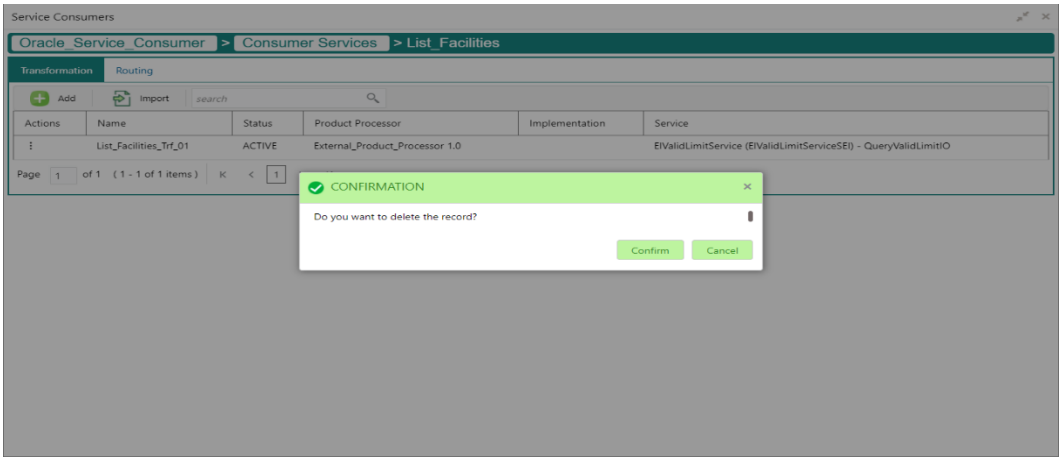

## 9.4 Edit

User can modify the transformation details.
Navigation: **Transformation -> Operation Menu (3 dot icon) -> Edit**



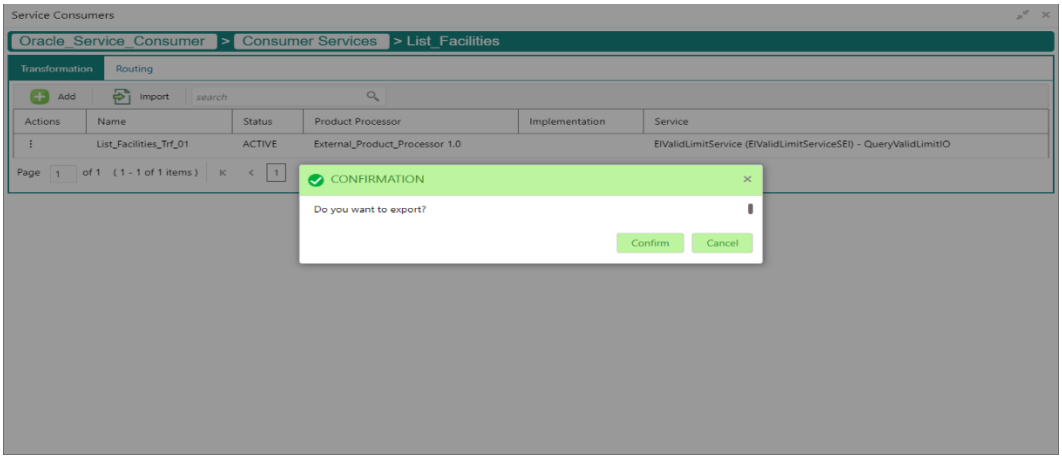

## 9.5 Delete

User can delete the transformation.
Navigation: **Transformation -> Operation Menu (3 dot icon) -> Delete**




## 9.6 Export

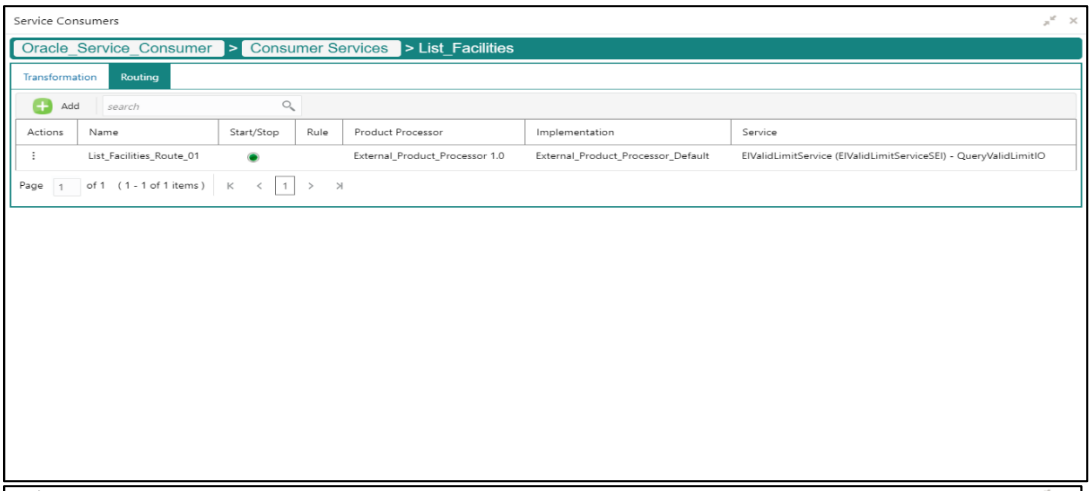
User can export the transformation configuration as JSON file.
Navigation: **Transformation -> Operation Menu (3 dot icon) -> Export**

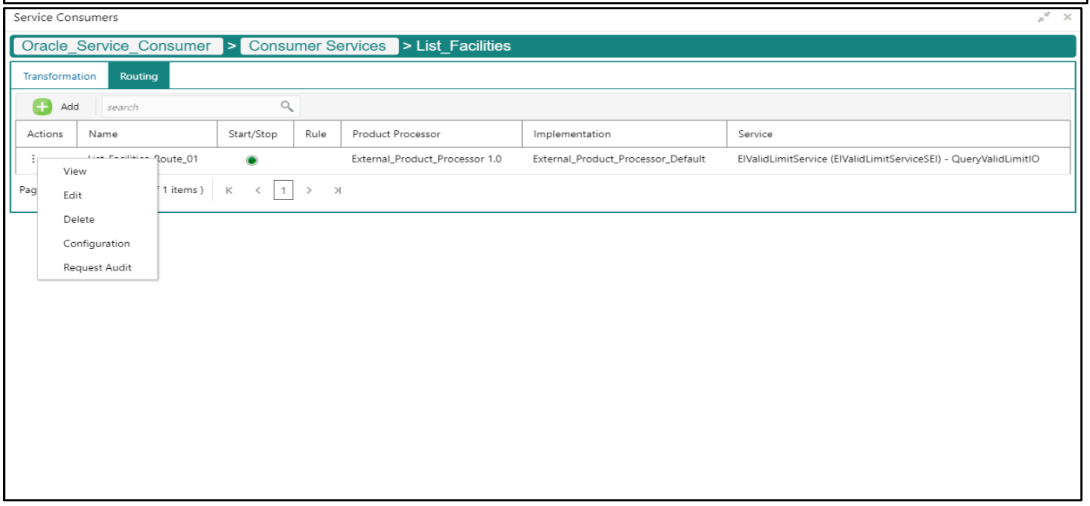


# 10. Routing

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

Navigation: **Core Maintenance -> Routing Hub -> Service Consumers -> <Specific Service Consumer> -> Consumer Services -> <Specific Consumer Service> -> Routing**



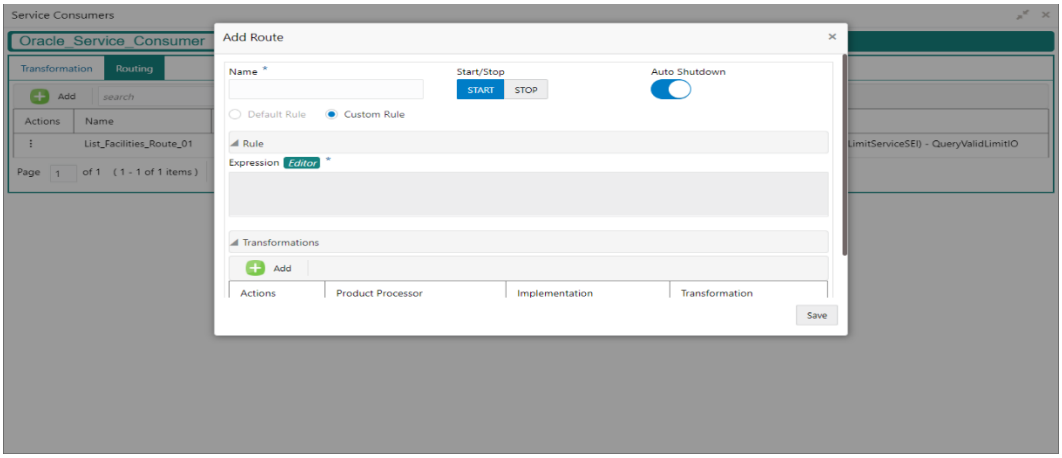




| **Component briefing** | | | |
|---|---|---|---|
| **Component Name** | **Component Type** | **Condition** | **Comments** |
| **<Service Consumer>** | Button | | Navigates back to Service Consumers |
| **<Consumer Service>** | Button | | Navigates back to Consumer Services |

| | | | |
|---|---|---|---|
| **Add** | Button | | Pops up add dialog |
| **Search** | Combo Box One | | Provides search functionality with case insensitive (Routing Name) |
| **Navigation: Routing -> 3 dot icon (operation menu)** | | | |
| **View** | menu option | Non-editable | Pops up view dialog |
| **Edit** | menu option | | Pops up edit dialog |
| **Delete** | menu option | | |
| **Configuration** | menu option | | Pops up configuration dialog |
| **Request Audit** | menu option | | Pops up request audit log |

## 10.1 Add

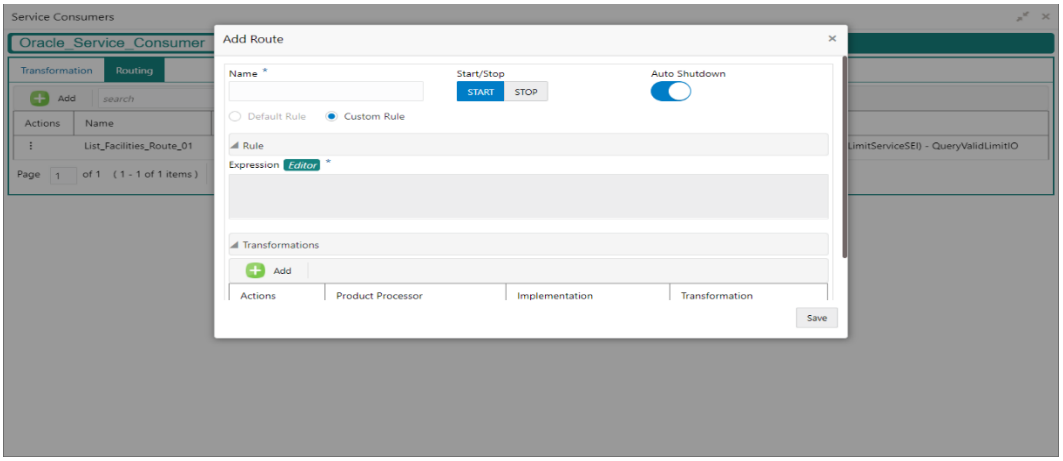
User can create Routing manually.
Navigation: **Routing -> Add**




| **Component briefing** | | | |
|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Comments** |
| Name | Text Box | Yes | Unique routing name |
| Start / Stop | Switch | Yes | Predefined Values: START / STOP<br>If routing is marked as STOP, then consumer request will fail at routing hub level only. |

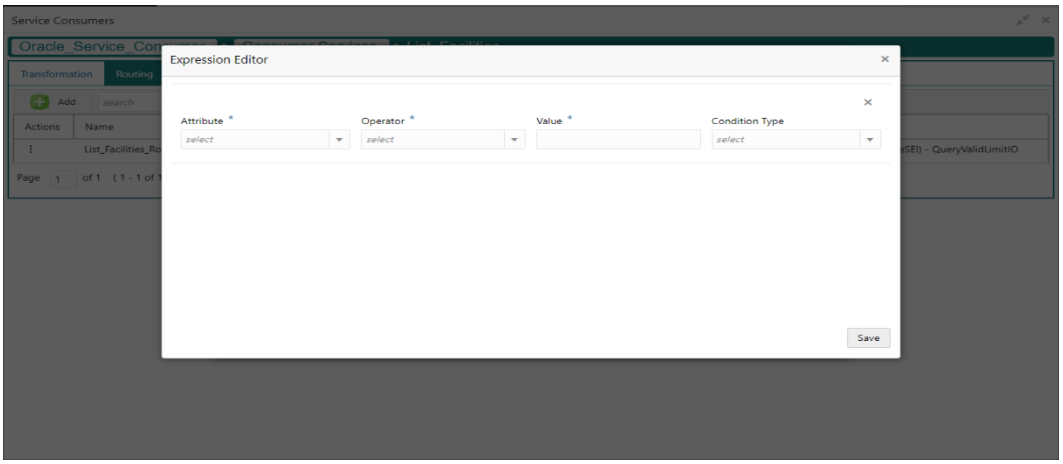
| | | | |
|---|---|---|---|
| Auto Shutdown | Switch | Yes | Predefined Values:<br>ON / OFF<br>If AutoShutdown flag is ON, then route state will be changed to STOP if route failure goes beyond the threshold failure limit based on the monitoring and alert configuration. |
| Rule Type | Radio Button | Yes | Predefined Values:<br>Default Rule / Custom Rule |
| Transformations | Collapsible Header & Content | | |
| Save | Button | | Saves the routing details |

### 10.1.1 **Add Routing with Custom Rule**



| Component briefing | | | | |
|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Data type** | **Comments** |
| Expression | Text Area | Yes | | User can view expression that is formed through expression editor. |
| Editor | Button | | | Pops up expression editor dialog |

## 10.1.2 Add Custom Rule using Expression Editor



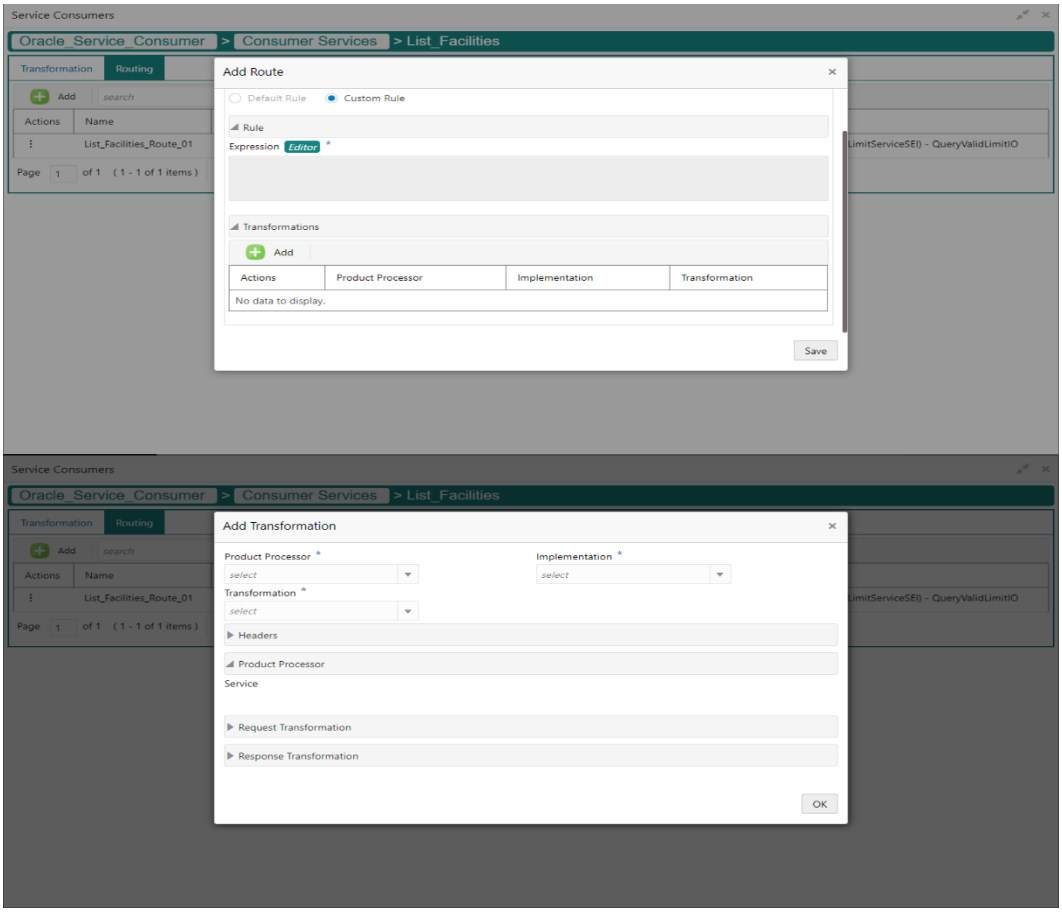

| Component briefing | | | |
|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Comments** |
| Attribute | Combo Box One | Yes | Displays list of attributes relevant to consumer service |
| Operator | Combo Box One | Yes | Logical operators to form an expression |
| Value | Text Box | Yes | |
| Condition type | Combo Box One | | Conditional Operators |

| | | | |
|---|---|---|---|
| Save | Button | | Saves the expression |

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

## 10.1.3 Transformations

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

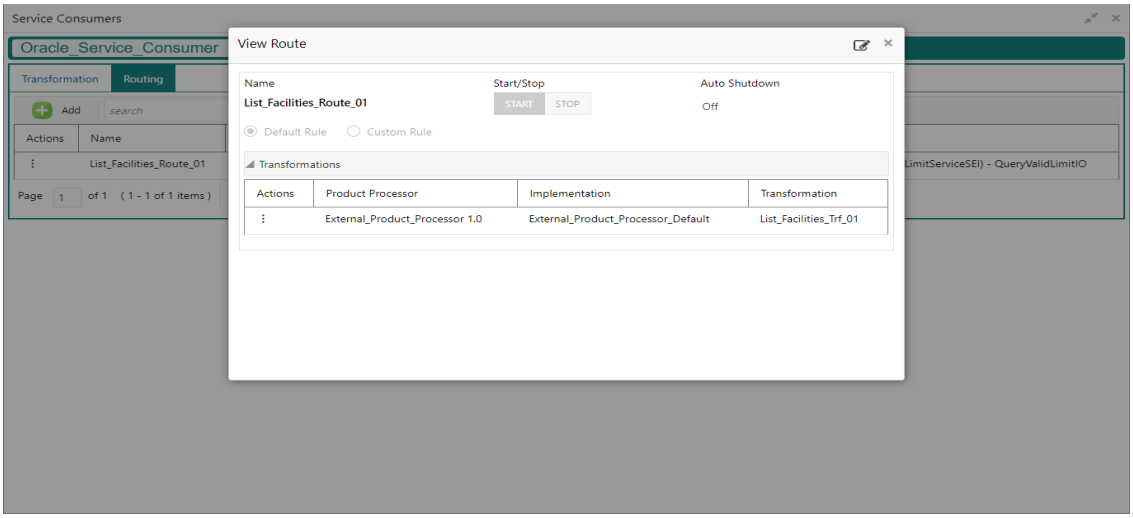



| Component briefing | | | |
|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Comments** |
| Product Processor | Combo Box One | Yes | Displays provider list relevant to consumer |

| | | | |
|---|---|---|---|
| Implementation | Combo Box One | Yes | Displays implementation list relevant to selected provider |
| Transformation | Combo Box One | Yes | Displays transformation list relevant to select provider & implementation |
| Headers | Collapsible Header & Content | | Displays header list relevant to selected implementation and transformation<br>User can change the header values.<br>Value can either be hardcoded or can be Velocity mapping. |
| Product Processor | Collapsible Header & Content | | Displays service details relevant to selected transformation |
| Request Transformation | Collapsible Header & Content | | Displays request transformation template |
| Response Transformation | Collapsible Header & Content | | Displays response transformation template |
| OK | Button | | Saves the transformation details in list |

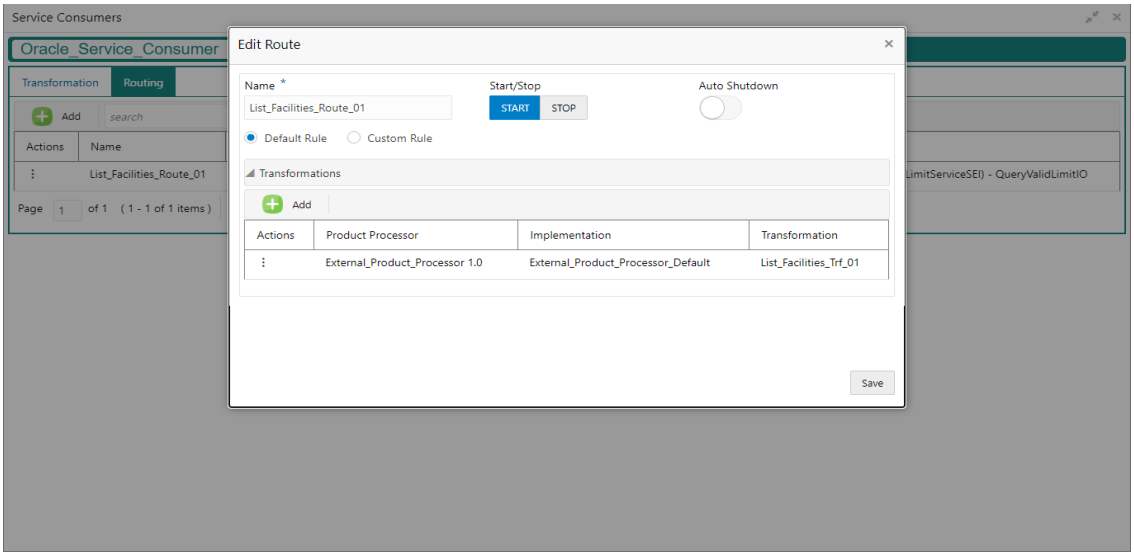
## 10.2 View

User can view routing details and can also switch to edit form by clicking on edit icon.
Navigation: **Routing -> Operation Menu (3 dot icon) -> View**



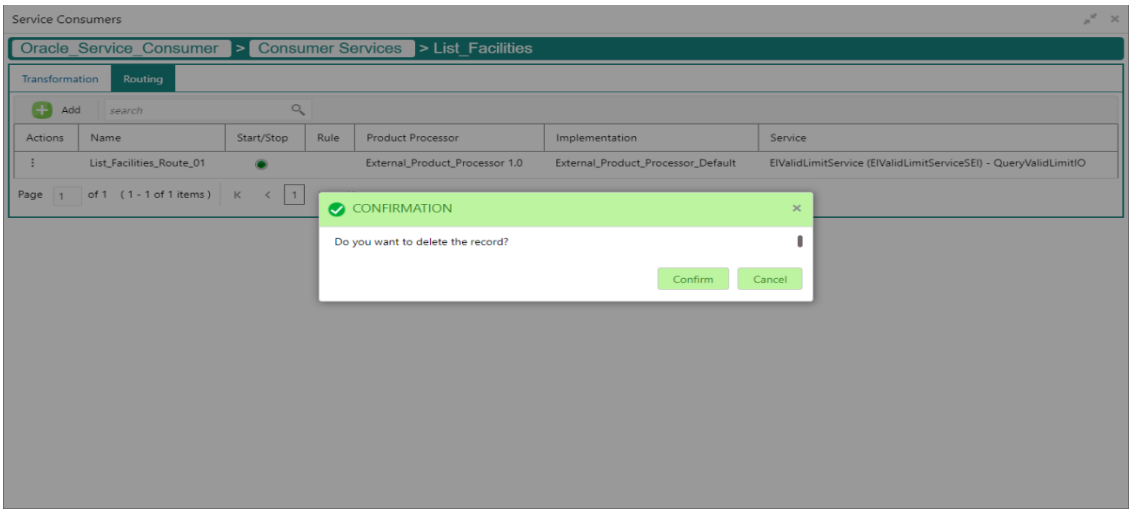

## 10.3 Edit

User can modify the routing details.
Navigation: **Routing -> Operation Menu (3 dot icon) -> Edit**




## 10.4 Delete

User can delete the routing.
Navigation: **Routing -> Operation Menu (3 dot icon) -> Delete**


Service Consumers
Oracle_Service_Consumer > Consumer Services > List_Facilities
Transformation
Routing
Add
search
Actions
Name
Start/Stop
Rule
Product Processor
Implementation
Service
List_Facilities_Route_01
External_Product_Processor 1.0
External_Product_Processor_Default
EIValidLimitService (EIValidLimitServiceSEI) - QueryValidLimitIO
Page 1 of 1 ( 1 - 1 of 1 items )
CONFIRMATION
Do you want to delete the record?
Confirm
Cancel

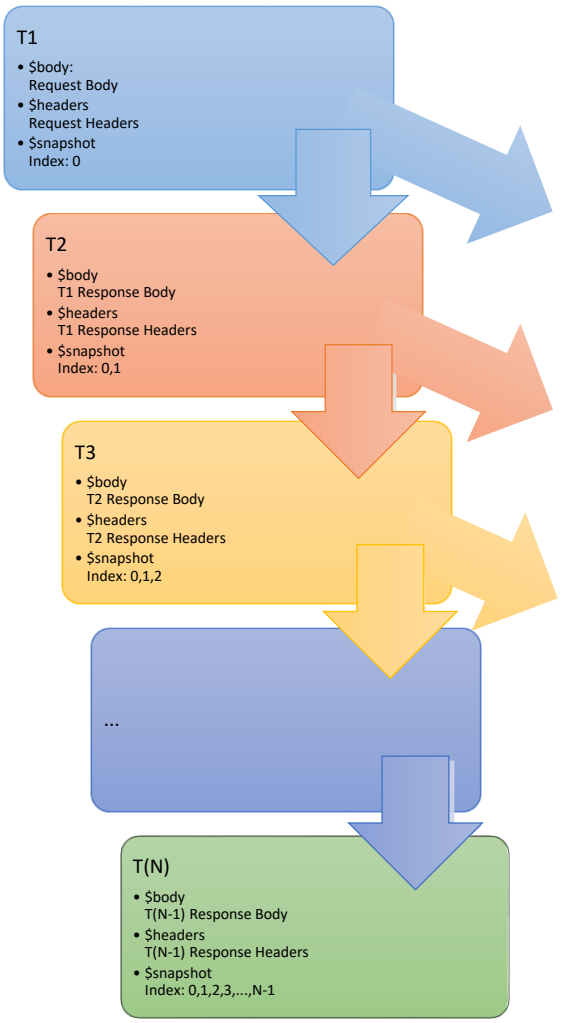
# 11. Chaining

End-user will be able to define the sequence of transformations for each routing in which request should be processed.

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

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

$body and $headers will refer the response body and headers of previous step.

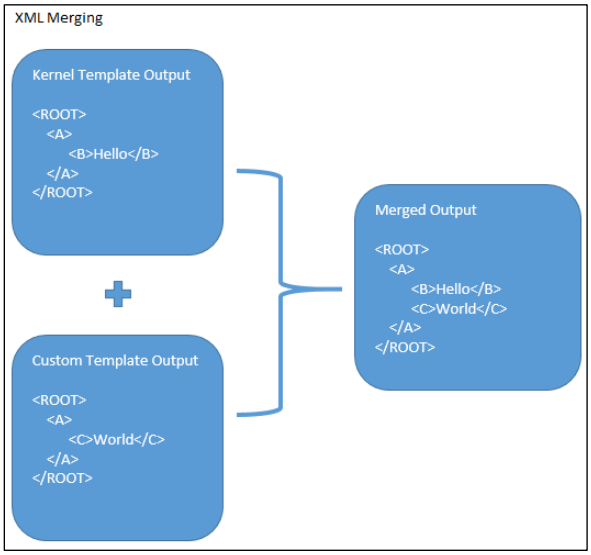


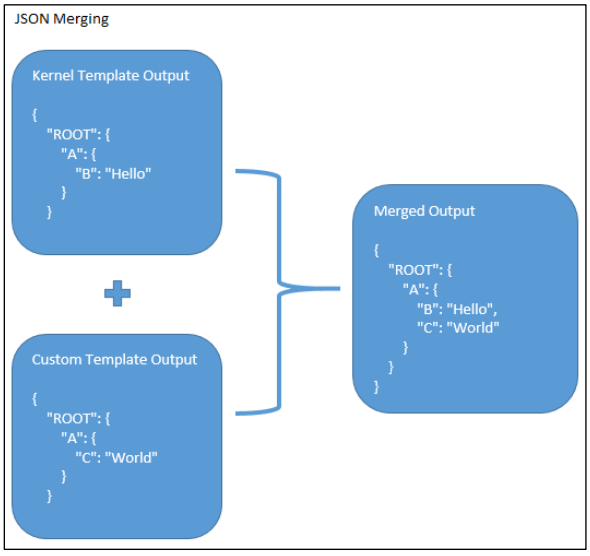

| **Snapshot List** | | |
|---|---|---|
| **Index** | **body** | **headers** |
| 0 | Request Body | Request Headers |
| 1 | T1 Response Body | T1 Response Headers |
| 2 | T2 Response Body | T2 Response Headers |
| 3 | T3 Response Body | T3 Response Headers |
| … | | |
| | | |

# 12. Extensibility

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

Example:



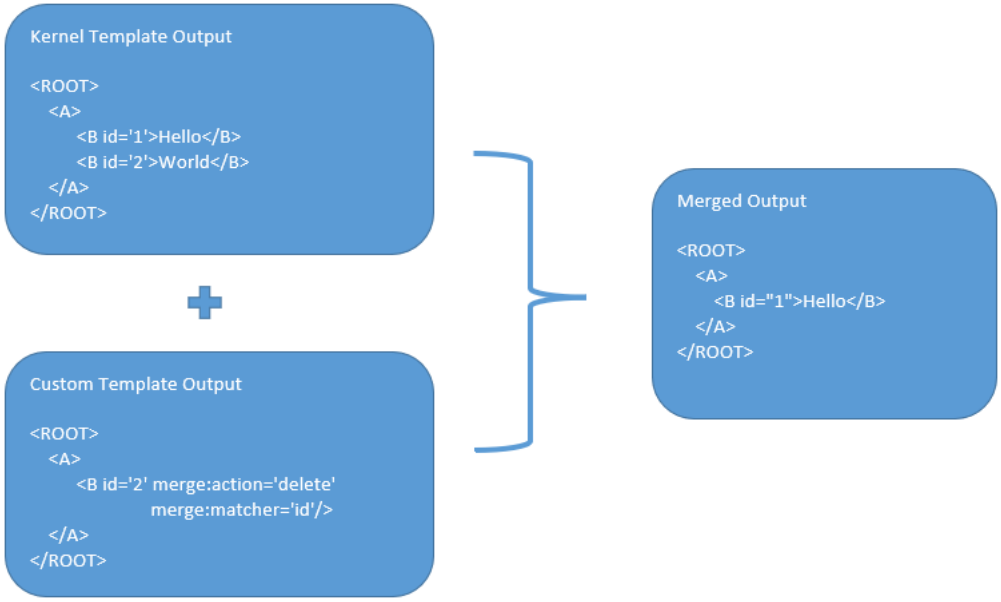
# 12.1 XML merging attributes

## 12.1.1 Match strategy

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

Syntax: merge:matcher='<ATTRIBUTE_NAME>'

Example:

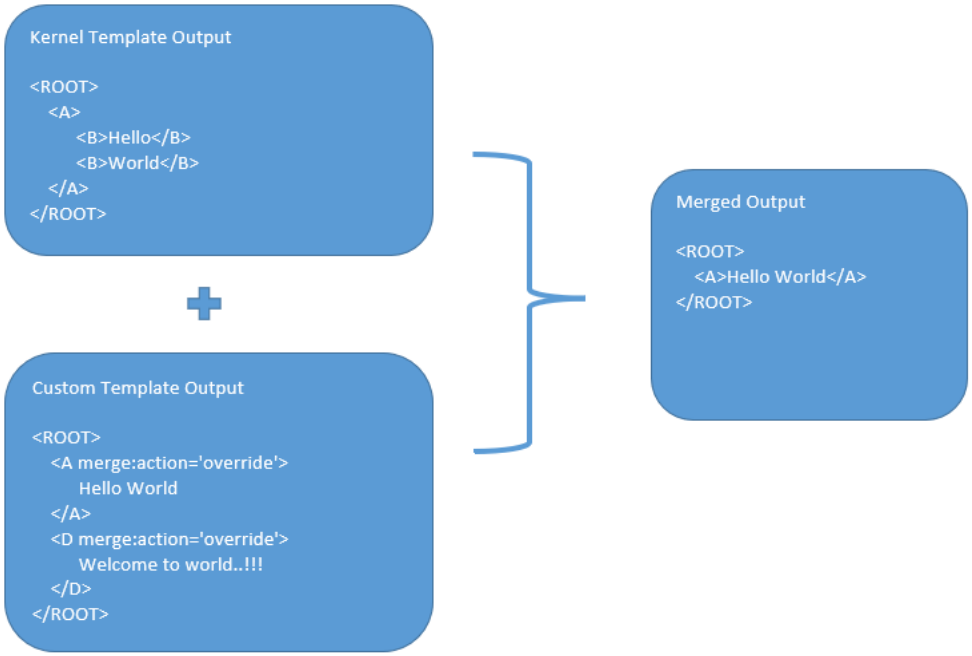



## 12.1.2 Override

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

Syntax: merge:action='override'

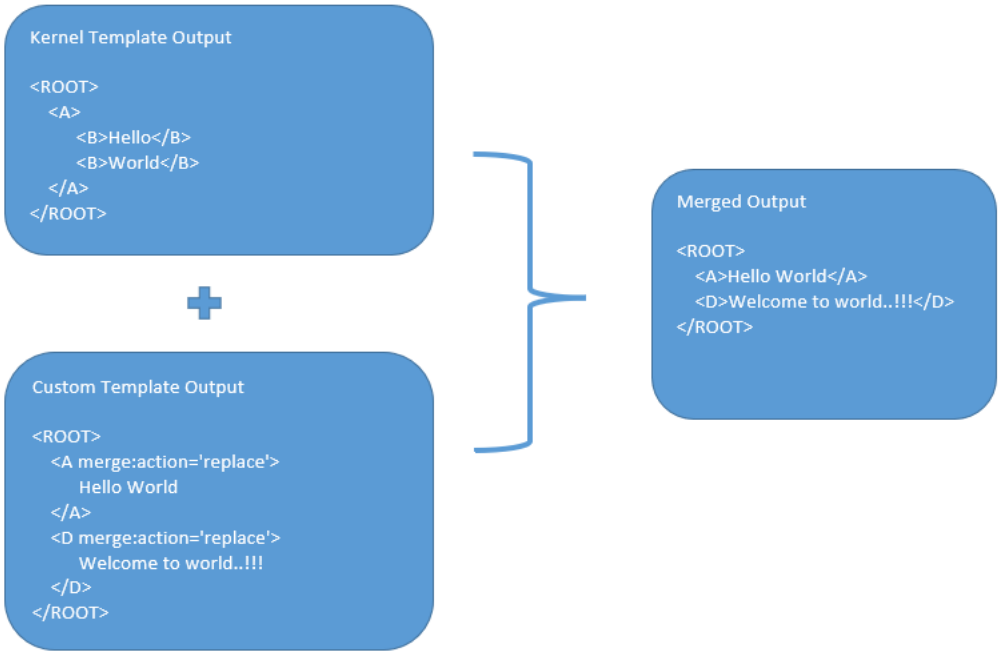
Example:




### 12.1.3 Replace

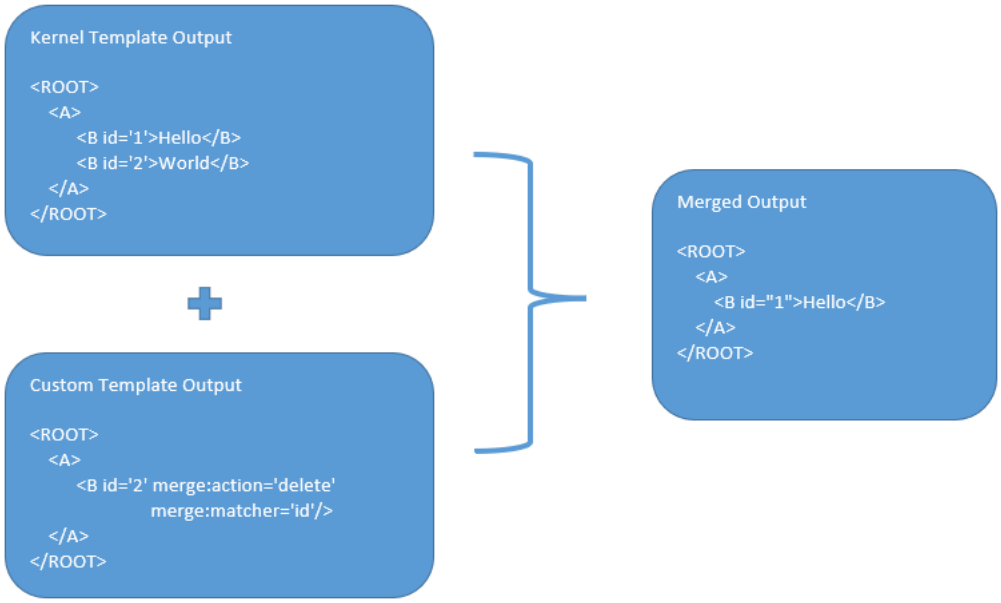
Replaces the original element with the patch element or creates the element if it does not exist in kernel/mock template.
Syntax: merge:action='replace'

Example:



### 12.1.4 Delete

Deletes the original element.
Syntax: merge:action='delete'
Example:



# 13. 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.

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

1. Open "Configure Tasks" screen (Task Management -> 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, 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.

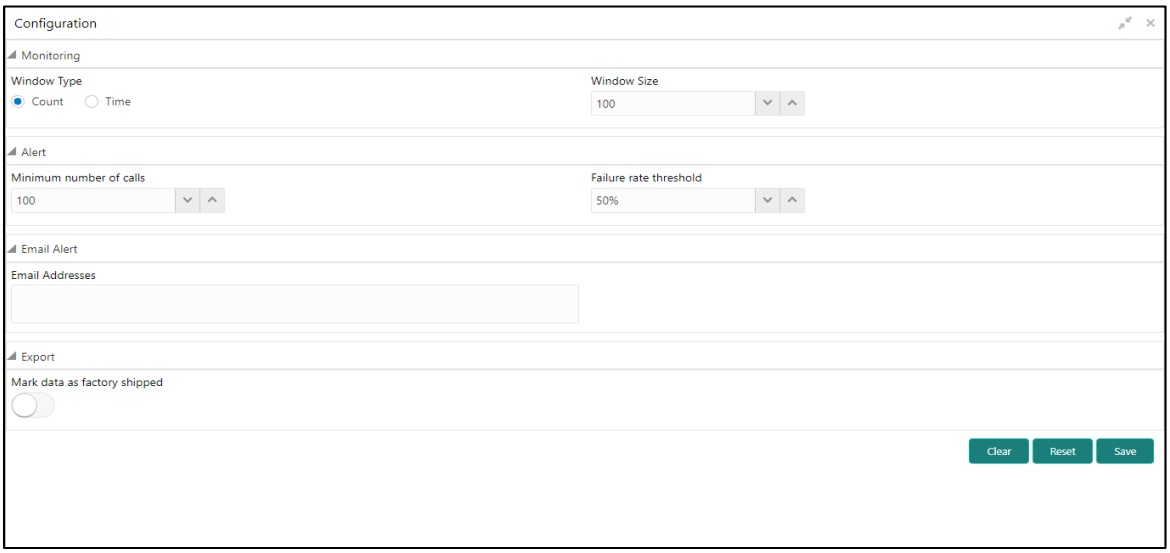
# 14. Multipart request

Below is the sample template for multipart request:

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

# 15. 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.
Navigation: **Core Maintenance -> Routing Hub -> Configuration**




| **Component briefing** | | | | |
|---|---|---|---|---|
| **Component Name** | **Component Type** | **Is Mandatory** | **Validation** | **Comments** |
| Monitoring | Collapsible Header & Content | | Monitoring properties are mandatory if alert properties have been configured. | This section has properties that are required by Breaker for storing and aggregating the outcome of calls using WindowType and WindowSize. |
| Window Type | Radio Button | No | | Predefined Values:<br>COUNT / TIME<br>The count-based sliding window aggregates the outcome of the last N calls (Window Size).<br>The time-based sliding window aggregates the outcome of the calls of the last N seconds (Window Size). |
| Window Size | Number Box | No | | This property is used to record the outcome of calls when the CircuitBreaker is closed. |

| | | | | |
|---|---|---|---|---|
| | | | | In case of count-based sliding window, window size will be N calls.<br>In case of time-based sliding window, window size will be N seconds. |
| Alert | Collapsible Header & Content | | Alert properties are mandatory if monitoring properties have been configured. | This section has properties that are required for transitioning CircuitBreaker. |
| Minimum number of calls | Number Box | No | | If minimumNumberOfCalls is 10, then at least 10 calls must be recorded, before the failure rate can be calculated.<br>If only 9 calls have been recorded the CircuitBreaker will not transition to open even if all 9 calls have failed. |
| Failure rate threshold | Number Box | No | | Configures the failure rate threshold in percentage.<br>When the failure rate is equal or greater than the threshold the CircuitBreaker transitions to open and starts short-circuiting calls. |
| Email Alert | Collapsible Header & Content | | | This section has properties that are required for mail notification. |
| Email Addresses | Text Box | No | | Once the failure rate crosses the failureRateThreshold, then end-user will be notified about the event via mail.<br>This property will be semi-colon separated email addresses |
| Export | Collapsible Header & Content | | | This section has properties that are required for exporting configuration JSON. This section will be visible at system level only. |
| Mark data as factory shipped | Switch | Yes | | This property is used to mark the exported configuration JSON as factory shipped JSON. |

| | | | | |
|---|---|---|---|---|
| | | | | So, end-user will not be able to modify or delete the certain data once imported.<br>Default value is false. |

**Example**

Let say

WindowType is Count

WindowSize is 20

Minimum number of calls is 10

Failure rate threshold is 50%

Configured properties will behave as below:

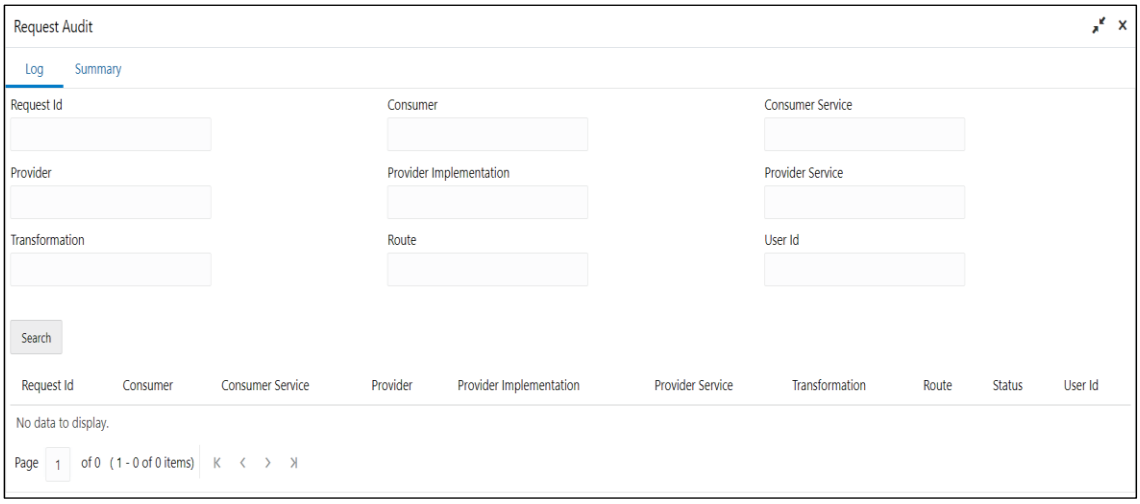
After 10 (min number of calls) calls, routing would get shutdown if 50% (failure rate) of atmost last 20 (window size) calls have failed.

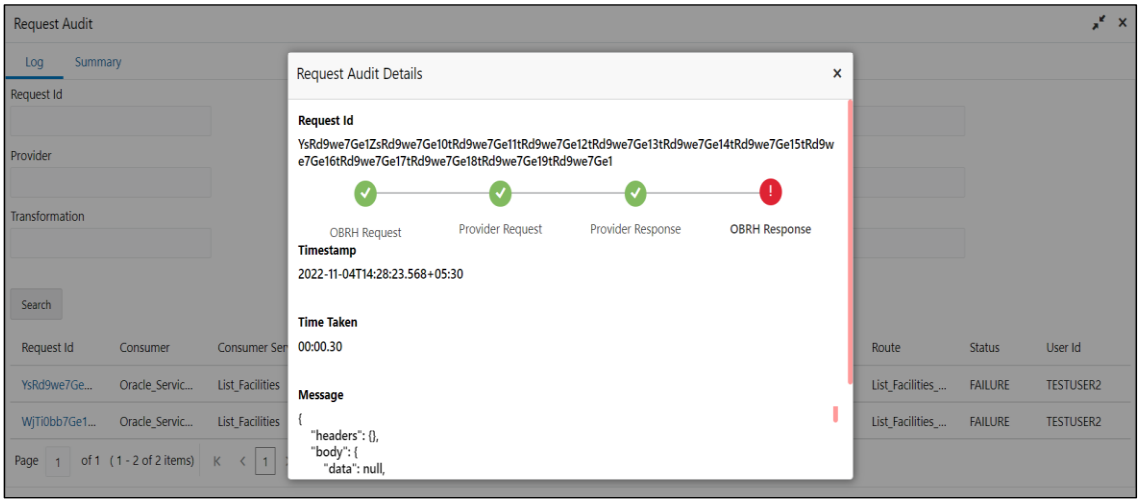
And if email address property is configured, then end-user will be notified as well.

# 16. Audit Log

Check audit log on the below navigation path for Oracle Banking Routing Hub

Navigation: **Core Maintenance -> Routing Hub -> Request Audit -> Log**



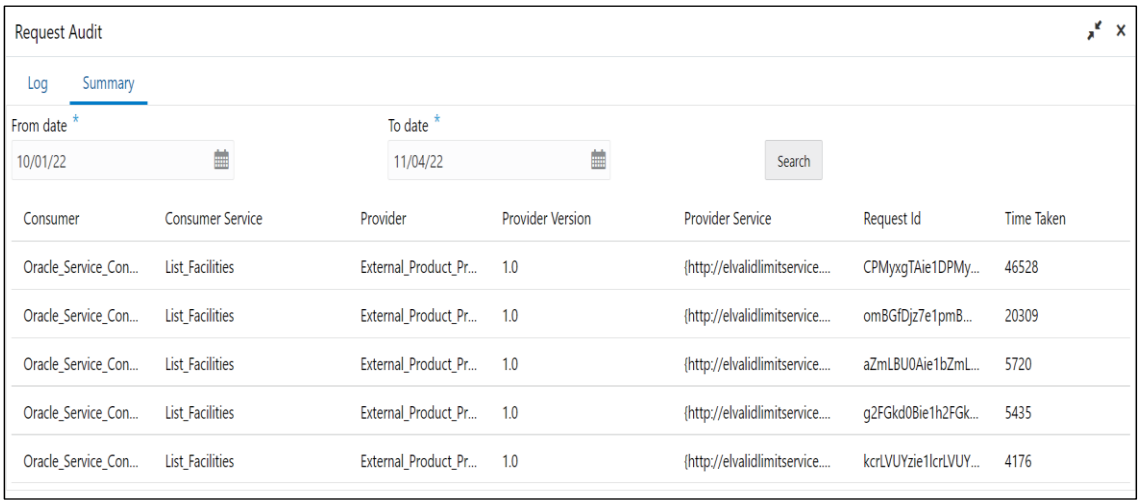
| Component briefing | | | |
|---|---|---|---|
| **Component Name** | **Component Type** | **Condition** | **Comments** |
| Consumer | Text Box | | Consume Name |
| Consumer Service | Text Box | | Service Consumer Name |
| Provider | Text Box | | Provider Name |
| Provider Implementation | Text Box | | Provider Implementation Name |
| Provider Service | Text Box | | Provider Service Name |
| Transformation | Text Box | | Transformation Name |
| Route | Text Box | | Route Name |
| Request Id | Text Box | | Correlation Id |
| User Id | Text Box | | User Id |
| Search | Button | Search is done based on below things:<br>• Case insensitive<br>• Pattern matching<br>• Single / Multi Column search | Performs search operation with specified values |

Clicking on “RequestId” value will display step by step execution of request with data such as Routing Hub Request & Timestamp, Provider Request & Timestamp, Provider Response & Timestamp, Routing Hub Response & Timestamp.

# 17. Audit Summary

Audit summary screen will show top 5 long running provider requests of each provider for specific date range.

Navigation: **Core Maintenance -> Routing Hub -> Request Audit -> Summary**




This screen requires summary job to be executed using plato-batch-server periodically basis.

Below steps are required to schedule summary job (routingHubAuditSummaryJob) once cmc-obrh-services and plato-batch-server is UP and RUNNING:

1. On **Home** Screen, Click **Task Management**
2. Under **Task Management**, Click "**Configure Tasks**"
3. Select "**Schedule**" option
4. Select "**Task Name**" as routingHubAuditSummaryJob and "**Task Trigger Name**" will be generated automatically.
5. Specify the CRON expression to daily EOD

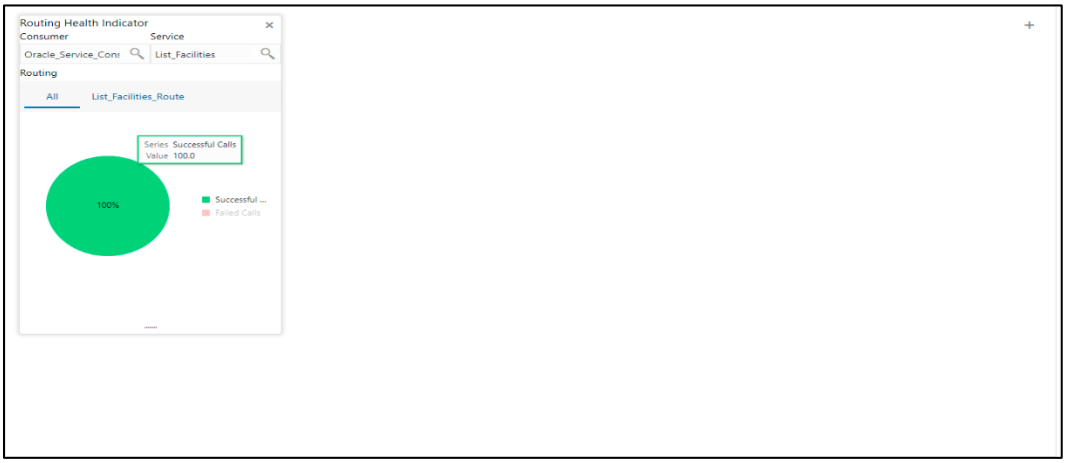
In order to resolve table space issue of Audit summary table, (CMC_RH_AUDIT_SUMMARY), Database Management Team has to configure database job to truncate table periodically basis.

# 18. Dashboard

## 18.1 Routing Health Indicator Widget

User can view the metric information Successful calls vs Failed calls ratio of each routing.

Failed calls here refer to the calls that are failed due to timeout issue.



# 19. Transformation Type

## 19.1 Velocity

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

In routing hub, Velocity will be used to generate JSON and XML.

- 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

    Please refer 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()

  **Please refer**

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

- Date Tool

  **Syntax:** $date.methodName()

  **Please refer**

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

- Json Tool

  **Syntax:** $json.methodName()

  **Please refer**

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

- Math Tool

  **Syntax:** $math.methodName()

  **Please refer**

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

- Number Tool

  **Syntax:** $number.methodName()

  **Please refer**

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

- Escape Tool

  **Syntax:** $esc.methodName()

  **Please refer**

  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:** List

- 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.

## 19.2 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 will be used to transform arbitrary XML to JSON.

## 19.3 JSLT

JSLT is a complete query and transformation language for JSON.

# 20. Oracle Banking Routing Hub Integration Specification

## 20.1 Token Generation

PlatoJWTAuth endpoint signature -

- Path : /platojwtauth
- Headers:
  - appId : SECSRV001
  - Content-Type : application/json
- Request Body:

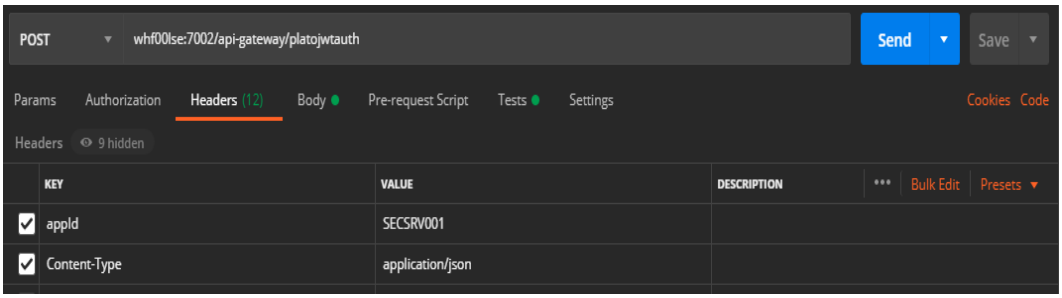
```
{
    "username": "",
    "password": ""
}
```

  - Username and password will be base64 encoding of plaintext.
- Response Body:

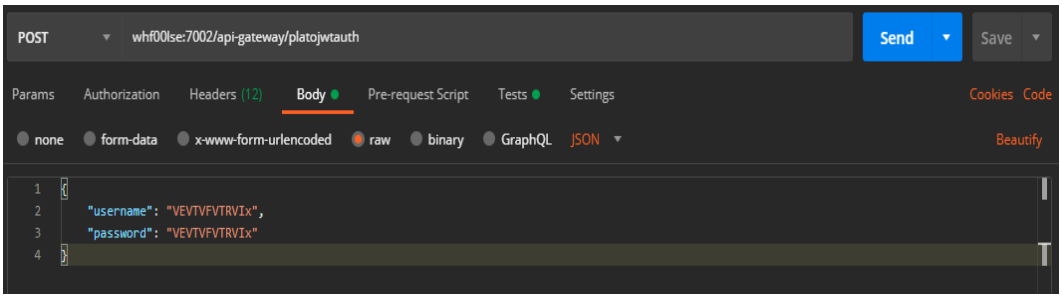
```
{
 "token": “”,
 "userAlreadyLoggedIn": “Y”,
 "expires_in": 3180,
 "home_entity_id": "DEFAULTENTITY",
 "multi_entity_admin": "N",
 "multi_entity_admin_locale": “”
}
```

Example - Below are the screenshots

**Headers**



**Request Payload**




**Response Payload**


## 20.2 Synchronous Dispatch API Specification

Dispatch endpoint is the **single** entry-point for invoking the routes configured in Oracle Banking Routing Hub for services of a Service Consumer.

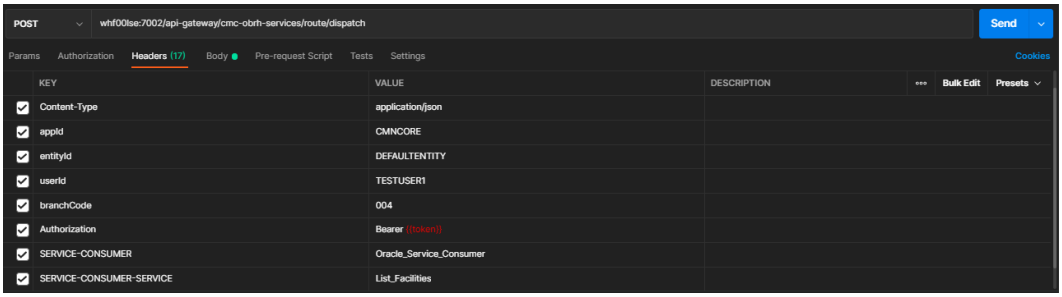
Dispatch endpoint signature -

- Method: POST / GET
- Path : /route/dispatch
- Headers:
  - appId : CMNCORE
  - entityId : DEFAULTENTITY
  - userId : <user id>
  - branchCode : <branch code>
  - Authorization : Bearer <Token>
  - SERVICE-CONSUMER : <name of service consumer>
  - SERVICE-CONSUMER-SERVICE : <name of service consumer service>
- Request Body:
  - Any valid JSON/XML payload which shall act as input to the transformation template in request transformer.
- Response Body:
  {
  "data": {}
  "messages": {}
  }

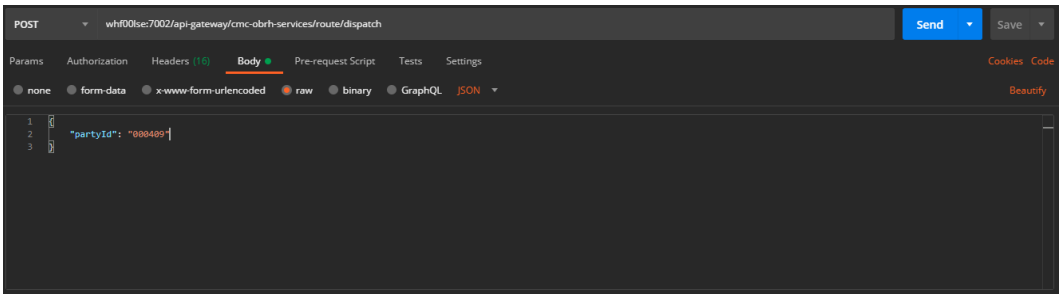
- If the route invocation succeeds, data JSON member would contain the transformed (optional) response of the provided service. If it's a provided web service and no response transformation template is supplied, XML Soap Body of response would be converted into JSON object and sent in data JSON member.
- If the route invocation fails due to misconfigured route or connection time out to ServiceProviderImpl or some other reason, relevant error messages would be sent in messages JSON member. In that case, data JSON member would be null or empty.

Example - Below are the screenshots of route dispatch for ServiceConsumer **Oracle_Service_Consumer** and ServiceConsumerService **List_Facilities**

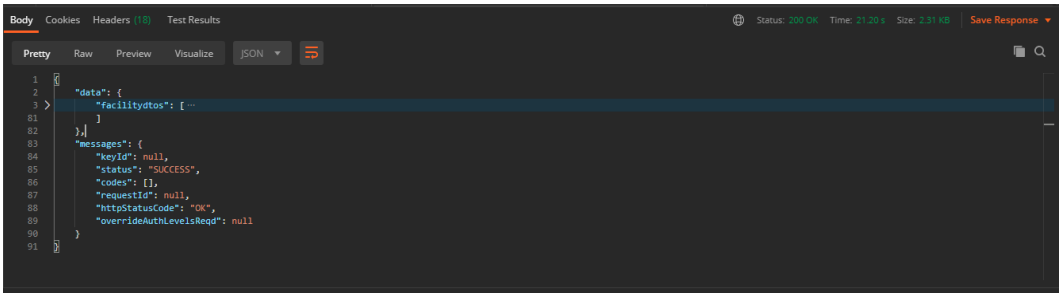
### Headers



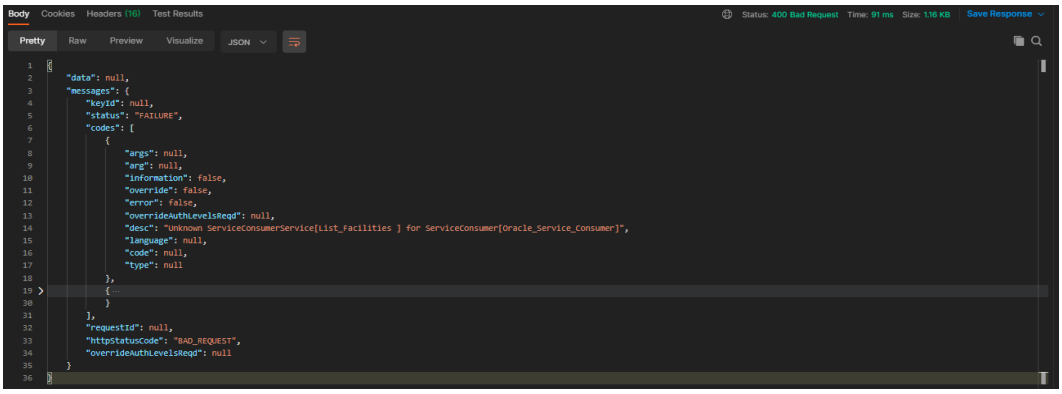

### Request Payload




### Response Payload on Successful Dispatch




### Response Payload on Failed Dispatch




## 20.3 Asynchronous Dispatch API Specification

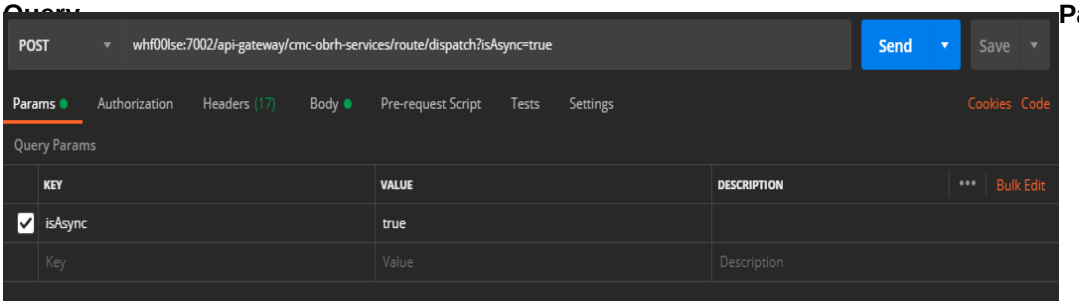
Dispatch endpoint is the **single** entry-point for invoking the routes configured in Oracle Banking Routing Hub for services of a Service Consumer.

Dispatch endpoint signature -

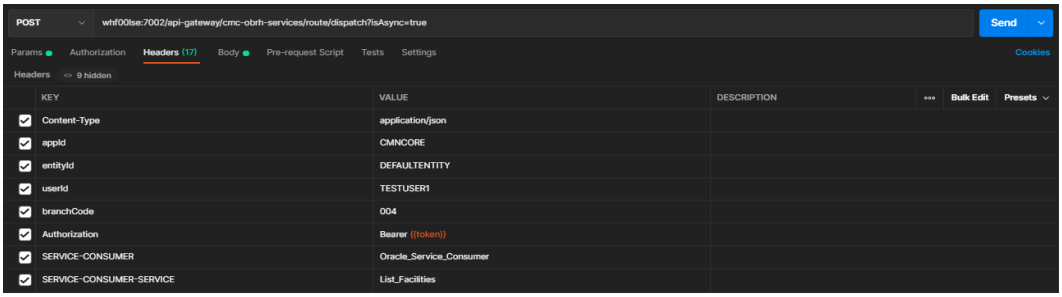
- Path : /route/dispatch
- Query Params:
  - isAsync : true
- Headers:
  - appId : CMNCORE
  - entityId : DEFAULTENTITY
  - userId : <user id>
  - branchCode : <branch code>
  - Authorization : Bearer <Token>
  - SERVICE-CONSUMER : <name of service consumer>
  - SERVICE-CONSUMER-SERVICE : <name of service consumer service>
- Request Body:
  - Any valid JSON/XML payload which shall act as input to the transformation template in request transformer.
- Response Body:
  {
  "data": { “correlationId” : “” },
  "messages": {}
  }

Example - Below are the screenshots of route dispatch for ServiceConsumer **Oracle_Service_Consumer** and ServiceConsumerService **List_Facilities**

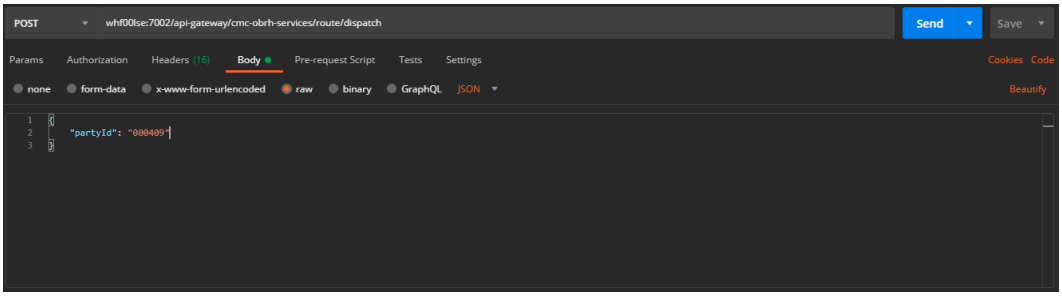
**Query Params**



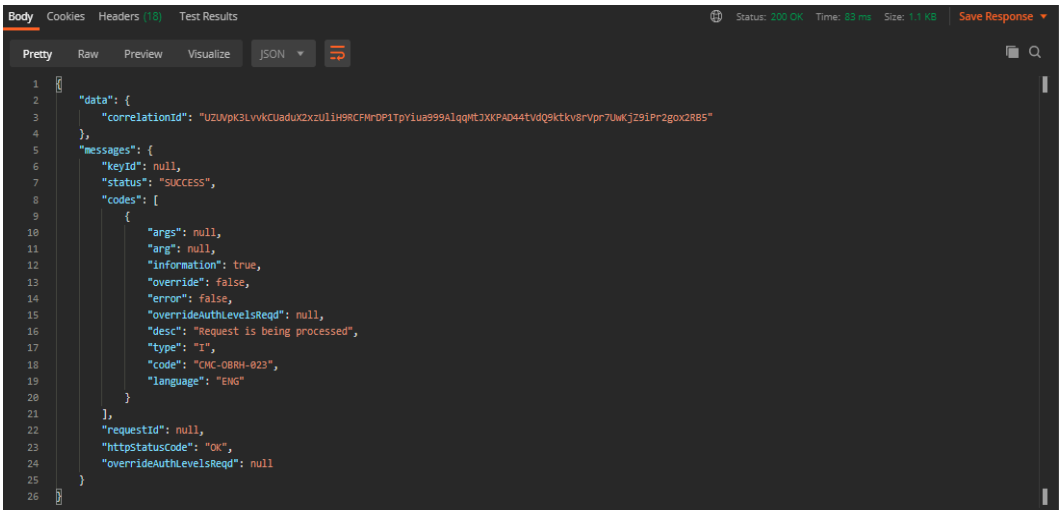

## Headers




## Request Payload



**Response Payload**



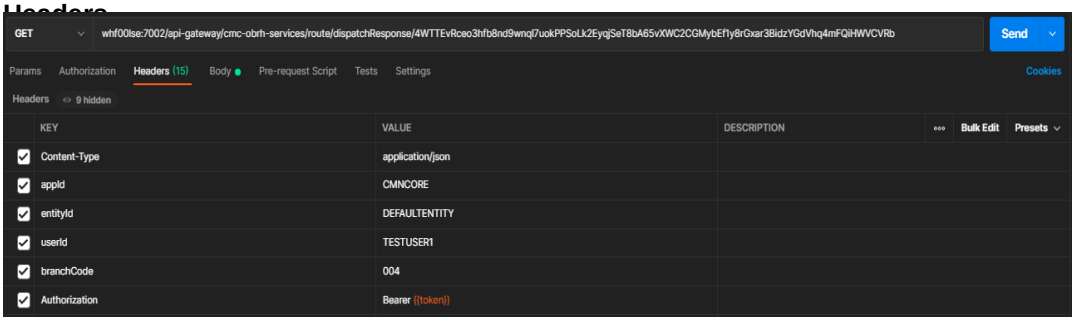

## 20.4 Asynchronous Dispatch Response API Specification

Response endpoint signature -

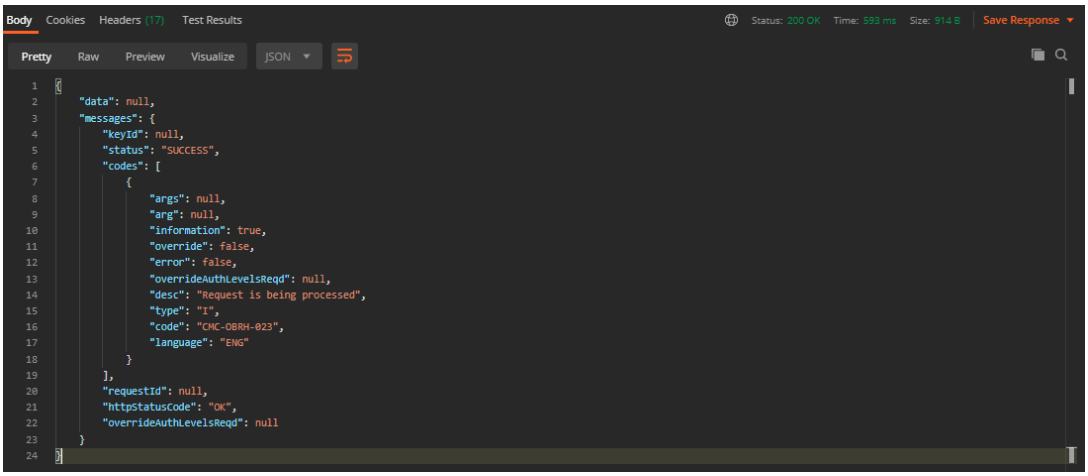
- Path : /route/dispatchResponse/{Correlation-Id}
  - Correlation-Id will be coming from the response of dispatch endpoint.
- Headers:
  - appId : CMNCORE
  - entityId : DEFAULTENTITY
  - userId : <user id>
  - branchCode : <branch code>
  - Authorization : Bearer <Token>
- Response Body:
  {
  "data": {}
  "messages": {}
  }
  - If the route invocation succeeds, data JSON member would contain the transformed (optional) response of the provided service. If it's a provided web service and no response transformation template is supplied, XML Soap Body of response would be converted into JSON object and sent in data JSON member.
  - If the route invocation fails due to misconfigured route or connection time out to ServiceProviderImpl or some other reason, relevant error messages would be sent in messages JSON member. In that case, data JSON member would be null or empty.

Example - Below are the screenshots of route dispatch for ServiceConsumer **Oracle_Service_Consumer** and ServiceConsumerService **List_Facilities**

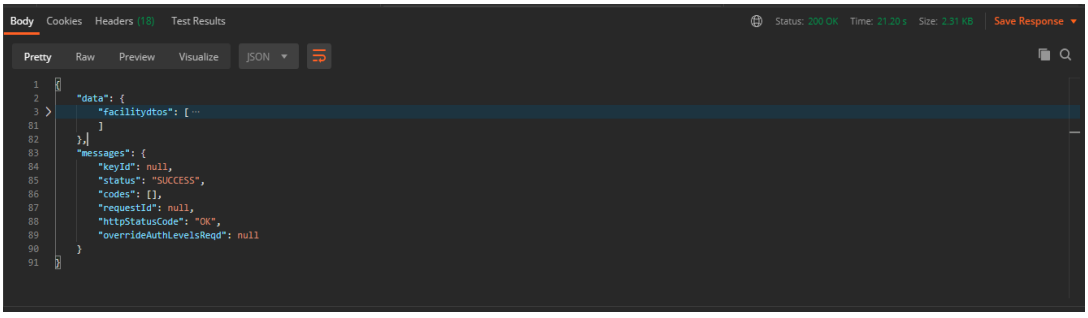
**Headers**



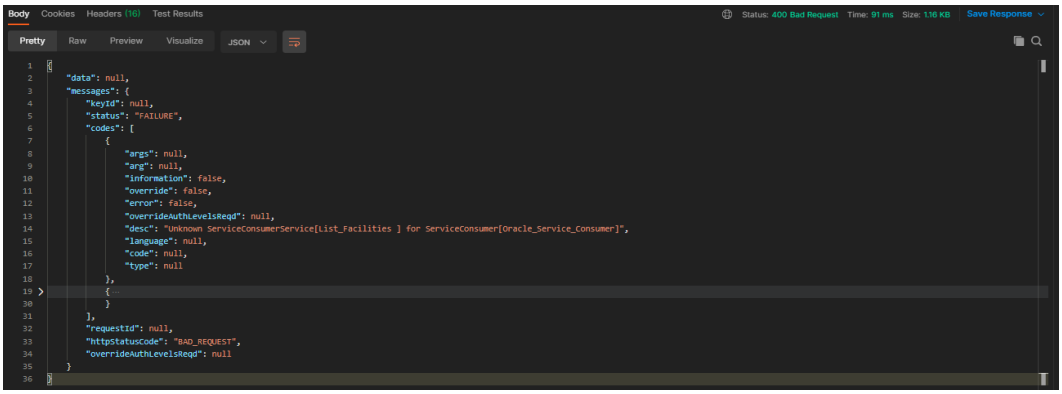

## Response Payload when request is still processing




## Response Payload when request is processed (on Successful Dispatch)



**Response Payload when request is processed (on Failed Dispatch)**




## 20.5 Template evaluation API Specification

Template evaluation endpoint will return the evaluated output of transformation template. End-user can validate the template based on the returned output.

Template evaluation endpoint signature -

- Method: POST
- Path : /template/evaluate
- Headers:
  - appId : CMNCORE
  - entityId : DEFAULTENTITY
  - userId : <user id>
  - branchCode : <branch code>
  - Authorization : Bearer <Token>
  - SERVICE-CONSUMER : <name of service consumer>
  - SERVICE-CONSUMER-SERVICE : <name of service consumer service>
  - SERVICE-TRANSFORMATION : <name of service transformation>
- Query Parameters:
  - transformationType : REQUEST / RESPONSE / MOCK_RESPONSE

  Note: Default value is REQUEST if not specified
- Request Body:
  - Any valid JSON / XML payload which shall act as input to the transformation template in request transformer.
- Response Body:

```
{
  "data": {}
  "messages": {}
}
```

# 21. Oracle Banking Routing Hub VM Arguments

**Common Core Managed Server**

-Dcmc-obrh-services.server.port=<SERVER_PORT>
-Dobrh.db.jndi=<CMNCORE_JNDI>

-Dcmc-obrh-services.oic.oauth.scope=<OIC_OAUTH_SCOPE>
-Dcmc-obrh-services.oic.secretStore.url=<OIC_SECRET_STORE_URL>
-Dcmc-obrh-services.oic.idcs.url=<OIC_IDCS_URL>
-Dcmc-obrh-services.audit.retention.days=<AUDIT_RETENTION_POLICY_DAYS>
-Dcmc-obrh-services.audit.retention.archival=<AUDIT_RETENTION_POLICY_APPROACH>
(Y for archiving and N for purging)

In order to receive routing failure mail notification via plato-alerts-management-service, then set the below Property to true, (Default value is false)
-Dobrh.alerts.enabled=<ALERTS_ENABLED>

In order to change the behavior of auditing, then set the below property from predefined values (DEFAULT / KAFKA / LOG / OFF), (Default value is DEFAULT)
-Dobrh.audit.type=<AUDIT_TYPE>

For KAFKA option, cmc-obrh-kafka-consumer service needs to be deployed.

In order to overwrite the customization that is not part of configuration json,
-Dobrh.import.overwrite=<IMPORT_OVERWRITE> (Default value is false)

In order to use Custom Keystore and Truststore for HTTPS scheme,
-Dobrh.keystore.password.encoded=<IS_PASSWORD_ENCODED> (true, if password is base 64 encoded)
-Dobrh.truststore.path=<TRUSTSTORE_PATH>
-Dobrh.truststore.password=<TRUSTSTORE_PASSWORD>
-Dobrh.usekeystore=<USE_KEYSTORE> (true, if keystore is required along with truststore)
-Dobrh.keystore.path=<KEYSTORE_PATH>
-Dobrh.keystore.password=<KEYSTORE_PASSWORD>
-Dobrh.keystore.alias=<KEYSTORE_ALIAS>
-Dobrh.keystore.aliaspassword=<KEYSTORE_ALIAS_PASSWORD>
-Dobrh.ssl.protocol=<SSL_PROTOCOL> (Default value is TLS)

In order to do tomcat deployment,
-Dobrh.server.isJavaEE=false (mandatory)
-Dobrh.taskexecutor.corepoolsize=<CORE_POOLSIZE> (default is 50) (optional)
-Dobrh.taskexecutor.maxpoolsize=<MAX_POOLSIZE> (default is 50) (optional)
-Dobrh.taskexecutor.queuecapacity=<QUEUE_CAPACITY> (default is 100) (optional)

In order to set Proxy settings for HTTPS,
-Dhttps.proxyHost=<PROXY_HOST_NAME>
-Dhttps.proxyPort=<PROXY_PORT>
-Dhttps.nonProxyHosts=<NON_PROXY_HOST_LIST>
-Dhttp.nonProxyHosts=<NON_PROXY_HOST_LIST>

Note: 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).

In order to set logger level,
- Dplato.service.logging.level=<LOG_LEVEL>

In order to do SSL based SOAP provider calls and if RoutingHub is deployed on weblogic environment,
-DUseSunHttpHandler=true

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

**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.
In order to import bigger files,
plato-api-gateway.multipart.max-file-size=<MAX_FILE_SIZE> (default is 1MB)
plato-api-gateway.multipart.max-request-size=<MAX_REQUEST_SIZE> (default is 10MB)

Note : -1 for no size constraint

Example:
plato-api-gateway.multipart.max-file-size=-1

plato-api-gateway.multipart.max-request-size=-1

**CMC-OBRH-KAFKA-CONSUMER**

-Dcmc-obrh-kafka-consumer.server.port=<SERVER_PORT>