

# Medius Connector

---



|                                                                                                  |            |
|--------------------------------------------------------------------------------------------------|------------|
| <b>Overview</b>                                                                                  | <b>2</b>   |
| <b>Introduction</b>                                                                              | <b>2</b>   |
| <b>Software Requirements</b>                                                                     | <b>2</b>   |
| <b>Connector Namespace and Schema</b>                                                            | <b>3</b>   |
| <b>Maven Dependency Information</b>                                                              | <b>3</b>   |
| <b>Configuration</b>                                                                             | <b>4</b>   |
| Creating a New Project                                                                           | 4          |
| Configuring the Medius Global Element                                                            | 5          |
| Authentication                                                                                   | 5          |
| Understanding the Medius Connector                                                               | 6          |
| <b>Common Use Cases</b>                                                                          | <b>7</b>   |
| External System Procurement Goods Receipt Note Confirmation and Search, Fetch, Update Status     | 7          |
| Export of Data - Image or Attachment                                                             | 27         |
| Procurement Purchase Order Confirmation and Cancel Confirmation and Search, Fetch, Update Status | 34         |
| External System Procurement Purchase Order Confirmation and Search, Fetch, Update Status         | 48         |
| Supplier Invoices Post Payment Information Messages and Search, Fetch, Update Status             | 73         |
| Master Data Import                                                                               | 85         |
| Send Payment Batch and Fetch Payment Batch Reconciliation Status and Payment Items               | 106        |
| <b>Resources</b>                                                                                 | <b>117</b> |

## Overview

The Medius connector allows you to access the Medius REST API through Anypoint Platform. The connector exposes Medius operations by executing their API calls as per configuration.

Medius connector has operations related to Master Data Import, Export of Data, Payment, Supplier Invoices, Procurement, External System Procurement Purchase Order and Goods Receipt Note modules. Medius connector has a total 52 operations.

Track features and API version updates using the Medius Connector Release Note. Review the connector operations and see how they work by reviewing the technical reference alongside the demo applications.

## Introduction

This document assumes that the user is familiar with [Medius](#), [Medius Success Portal](#), [Medius Access API](#), [Anypoint Connectors](#) and [Anypoint Studio](#). To increase familiarity with Studio, consider completing a [Anypoint Studio Tutorial](#). This page requires basic knowledge of [Mule Key Concepts](#) and [Medius](#), [Medius Success Portal](#).

## Software Requirements

For software requirements, visit the [Medius Connector Release Note](#).

## Installation

To install the connector in Anypoint Studio use the instructions in [Add Modules to Your Project](#).

Additionally, we recommend keeping Studio up to date with its latest version.

## Connector Namespace and Schema

While designing a Mule application in Anypoint Studio, when a user drags the connector from the palette onto the Anypoint Studio canvas, the studio automatically populates the XML code with the connector namespace and schema location.

|                        |                                                                                                                                                                                                                                                                                                                                                            |
|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Namespace</b>       | <a href="http://www.mulesoft.org/schema/mule/mule4-medius-connector">http://www.mulesoft.org/schema/mule/mule4-medius-connector</a>                                                                                                                                                                                                                        |
| <b>Schema Location</b> | <a href="http://www.mulesoft.org/schema/mule/mule4-medius-connector">http://www.mulesoft.org/schema/mule/mule4-medius-connector</a><br><a href="http://www.mulesoft.org/schema/mule/mule4-medius-connector/current/mule-mule4-medius-connector.xsd">http://www.mulesoft.org/schema/mule/mule4-medius-connector/current/mule-mule4-medius-connector.xsd</a> |

If you are manually coding the Mule application in Studio's XML editor or another text editor, define the namespace and schema location in the header of your Configuration XML, inside the <mule> tag.

```
<mule
  xmlns:http="http://www.mulesoft.org/schema/mule/http" xmlns:mule4-medius-
  connector="http://www.mulesoft.org/schema/mule/mule4-medius-connector"
  xmlns="http://www.mulesoft.org/schema/mule/core"
  xmlns:doc="http://www.mulesoft.org/schema/mule/documentation"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.mulesoft.org/schema/mule/core
    http://www.mulesoft.org/schema/mule/core/current/mule.xsd
    http://www.mulesoft.org/schema/mule/mule4-medius-connector
    http://www.mulesoft.org/schema/mule/mule4-medius-connector/current/mule-mule4-medius-connector.xsd
    http://www.mulesoft.org/schema/mule/http
    http://www.mulesoft.org/schema/mule/http/current/mule-http.xsd">

</mule>
```

**Note:** Use current in the schema path. Studio interprets this to the current Mule version.

## Maven Dependency Information

After downloading and installing the connector, following steps make the Medius connector available inside a Mule application for use and to package the application with the connector. If using Anypoint Studio, it will do this automatically. For Maven dependency management, include this XML snippet in the pom.xml

file in the Mule project.

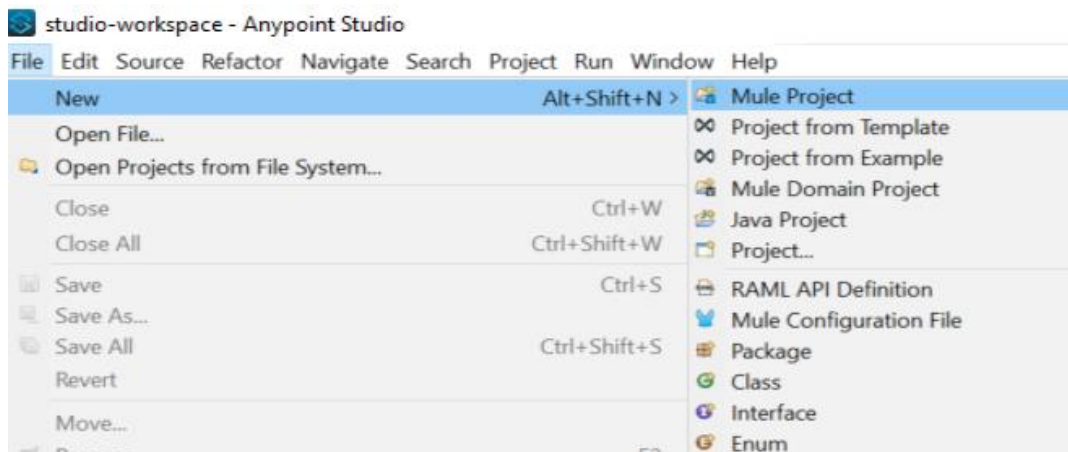
```
<dependency>
  <groupId>com.mulesoft.connectors</groupId>
  <artifactId>mule4-medijs-connector</artifactId>
  <version>1.0.0</version>
  <classifier>mule-plugin</classifier>
</dependency>
```

## Configuration

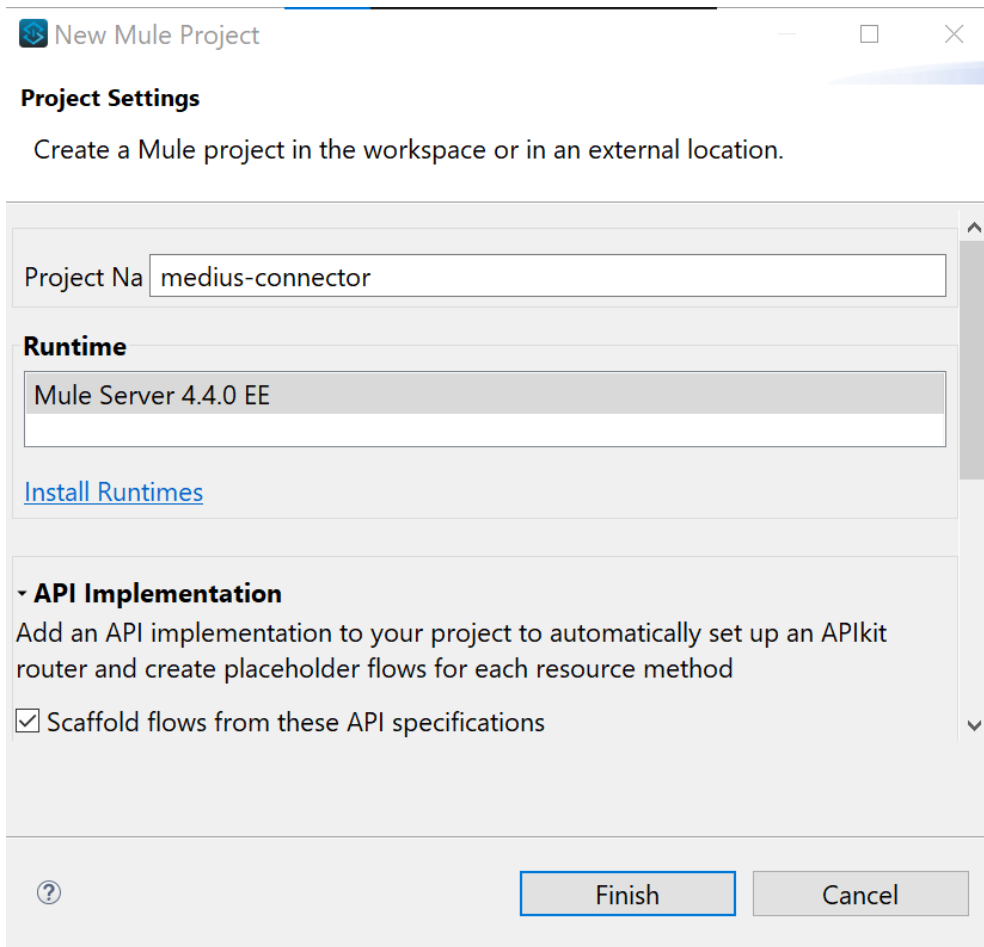
### Creating a New Project

To use the Medius connector in a Mule application project:

1. In Anypoint Studio, click File > New > Mule Project. Select Mule project from the dialog box.



2. Enter project name and specify Runtime, API implementation and Project Location if needed.



3. Click **Finish** to create the project.

## Configuring the Medius Global Element

To use the Medius connector in the Mule application, users must configure a global Medius element that is used by the Medius connector to authenticate. The Medius connector supports Client credentials authentication. Please check [Medius Document](#) for further information.

## Authentication

To access the UiPath API, the user needs to configure as mentioned below.  
Following parameters are required for CLIENT CREDENTIALS configuration:

| Field | Description |
|-------|-------------|
|-------|-------------|

|                      |                                                                                                                                                                                                                                                                            |
|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Base Url</b>      | Medius base Url -<br>( <a href="https://api.medius.com/">https://api.medius.com/</a> <tenant_name>)                                                                                                                                                                        |
| <b>Client Id</b>     | The OAuth Client Id - as registered with the service provider. Please refer to <a href="#">Application Setup</a> for further reference.                                                                                                                                    |
| <b>Client Secret</b> | The OAuth Client secret - as registered with the service provider.                                                                                                                                                                                                         |
| <b>Token Url</b>     | The service provider's token endpoint Url. refer <a href="#">Medius Document</a> for further information.<br>For example - ( <a href="https://login.medius.com/">https://login.medius.com/</a> <tenant>/connect/token)                                                     |
| <b>Scopes</b>        | The OAuth scope to be requested during the dance.<br>Values will be space separated.<br><br>Following is the minimum scope required for all operations to work:<br>Integration.FileExport Integration.DocumentImport Integration.Erp<br>Integration.Pay.PaymentBatchImport |

 Global Element Properties ✕

**Medius Connector Mule4 Config**

General Advanced Notes Help

Name: Medius\_Connector\_Mule4\_Config

Connection

General Proxy Security Advanced

Base Uri:  https://api.medius.com/<tenant>

oauthClientCredentials

Client id:  <Client Id>

Client secret:  <Client Secret>

Token url:  https://login.medius.com/<tenant>/connect/token

Scopes:  .<Scopes Space Separated>

oauthStoreConfig

Object store: 

▼



OK

Cancel

## Understanding the Medius Connector

The Medius connector functions within a Mule application. Medius connector allows you to perform Medius server operations through mule applications. Building an application that connects with Medius the user doesn't have to go through the effort of custom-coding (and securing!) a connection. Rather, just drop a connector into the Mule flow, configure a few connection details and begin an application running on Medius.

The Medius connector is configured to authorize access to Medius operations. It requires Base URL, Client Id, Client Secret, Token URL, Scopes for the authentication using Client Credentials flow.

The real value of the Medius connector is in the way you use it at design-time in conjunction with other functional features available in Mule.

- **DataSense:** DataSense extracts metadata for Medius standard response to automatically determine the data type and format that your application must deliver to, or can expect from, Medius server. Mule does the heavy lifting of discovering the type of data you must send to, or be prepared to receive from Medius server.
- **Transform Message Component:** This component's integrated scripting language called DataWeave can automatically extract response metadata that you can use to visually map and/or transform to a different data format or structure. Essentially, DataWeave lets you control the mapping between data types. For example, if you configure a Medius connector in your application, then drop a Transform Message component after the connector, the component uses DataWeave to gather information that DataSense extracted to pre-populate the input values for mapping. In other words, DataSense makes sure that DataWeave knows the data format and structure it must work with so you don't have to figure it out manually. Mule developers can refer to Medius Connector API docs and Medius Documentation for more information about complex input or return types.

## Common Use Cases

- External System Procurement Goods Receipt Note Confirmation and Search, Fetch, Update Status
- Export of Data - Image or Attachment
- Procurement Purchase Order Confirmation and Cancel Confirmation and Search, Fetch, Update Status
- External System Procurement Purchase Order Confirmation and Search, Fetch, Update Status
- Supplier Invoices Post Payment Information Messages and Search, Fetch, Update Status
- Master Data Import
- Send Payment Batch and Fetch Payment Batch Reconciliation Status and Payment Items

## External System Procurement Goods Receipt Note Confirmation and Search, Fetch, Update Status

This use case demonstrates the Search, Fetch and Update and Bulk Update flow as well as External System Procurement Goods Receipt Note Confirmation or Post Procurement Error flow based on selection of flow to execute.

- In Anypoint Studio, click **File > New > Mule Project**, name the project **mule-medi-us-connector-external-sys-goods-receipt-note-demo** and click Finish.
- In the search field, type **Http** and drag the **HTTP Listener** to the canvas.

- Click the **HTTP Listener**, click the green plus sign to the right of Connector Configuration. On the next screen, add the host and port, click OK
- Provide **Path** to HTTP Listener (Example: **/external-sys-goods-receipt-notes**)
- On the Mule Palette, search for **Set Variable** and drag it onto the canvas. Set value of variable from **mule-application.properties** file key “**selected.option**” as **\${selected.option}** and give name of variable as **selectedOption** and display name as **Selected Option**.
- There are two options available for execution of two different flows. This option user can change from **mule-application.properties** file.
  - **MSG\_SEARCH\_FETCH\_UPDATE** - execute Search, Fetch Status, Fetch Content, Update Status and Bulk Update Status operations related flow
  - **CONFIRMATION** - Send External System Procurement Goods Receipt Note record to ERP and then Confirmation flow.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- There will be two **When** blocks for added **Choice**. Those will be as follows.
- Add below condition as expression in **Choice** to check if user wants to execute Search, Fetch Status, Fetch Content, Update Status and Bulk Update Status operations related flow.

```
vars.selectedOption == "MSG_SEARCH_FETCH_UPDATE"
```

- Add **Flow Reference** to call another sub flow '**external-sys-goods-receipt-note-message-flow**' to Search, Fetch Status, Fetch Content, Update Status and Bulk Update Status operations related flow.
- Add below condition as expression in **Choice** to check if the user wants to execute External System Procurement Goods Receipt Note Confirmation operation related flow.

```
vars.selectedOption == "CONFIRMATION"
```

- Add **Flow Reference** to call another sub flow '**external-sys-goods-receipt-note-confirmation-flow**' to execute Send Record to ERP and then External System Procurement Goods Receipt Note Confirmation operation related flow
- Create a new sub flow **external-sys-goods-receipt-note-message-flow**.
- Add **Set Variable** and set variable name as **searchStatus** and value as **\${search.status}** key from **mule-application.properties** file and display name as **Search Status**.
- Add **Search External System Goods Receipt Notes** Medius operation from **Mule Palette** to canvas and provide **status** as expression **upper(vars.searchStatus)**. This operation returns fetched message status the parameters fetched in. The mandatory parameter is the status parameter, besides that the other parameters are fetched in as query string parameters to narrow down the

result.

- Add **Logger** to log searched records and set message as expression **payload**
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if payload is not null.

`payload != Null`

- Add **Set Variable** and set variable name as **pos** and value as **\${message.element.position}** key from **mule-application.properties** file and display name as **Set Position**. It will say which record user wants to process further based on position. For the first record its value will be 0, for the second record it will have 1 and so on.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if selected position is greater than or equal to zero and size of payload has record for that particular position.

`vars.pos >= 0 and sizeof(payload) > vars.pos`

- Add **Set Variable** and set variable name as **elementValue** and value as expression **output application/json --- payload[vars.pos as Number]** and display name as **Get and Set Element Value**. It will fetch the element from a particular position.
- Add **Set Variable** and set variable name as **link** and value as expression **vars.elementValue.url splitBy ("/")** and display name as **Set Link Split By /**. It will be useful to set a message Id.
- Add **Set Variable** and set variable name as **messageId** and value as expression **vars.link[-1]** and display name as **Set Message Id** here -1 means last element from array and it is messageId.
- Search **Get Goods Receipt Note Message Status** Medius operation from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId**. The operation with message id as input will return a status and an effective link to the real goods receipt note message status. The main idea with this method is to be able to check status for a given message and to be able to find where a message is in terms of statuses.
- Add **Set Variable** and set variable name as **fetchStatus** and value as expression **upper(payload.status)** and display name as **Set Fetched Status**
- Add **Set Variable** and set variable name as **newStatus** and value as **\${new.update.status}** from mule properties file and display name as **New Update Status**. This indicates a new update status for a given record. It will be useful in further operations.
- Search **Get Goods Receipt Note Message Content** from **Mule Palette** and drag to canvas and provide

Message Id as expression **vars.messageId** and Message Status as expression **upper(vars.fetchedStatus)**

This operation returns the status good note receipt content from a message that matches the provided status and id. If the message id is in another status, nothing is returned.

- Add **Logger** to log response and set message as expression **payload**
- Add **Set Variable** and set variable name as **corKey** and value as expression **payload.CorrelationKey** and display name as **Set Correlation Key**
- Search **Update Goods Receipt Note Status** from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId** and Message Status as expression **vars.fetchedStatus** and provide **body** parameter as expression as follows:

```
%dw 2.0
output application/json
---
{
  "NewStatus": vars.newStatus
}
```

- This operation updates the status for an updated goods receipt note status when processing it.
- Add **Logger** to log response and set message as expression **payload**
- Search **Bulk Update External System Goods Receipt Notes Statuses** from **Mule Palette** and drag to canvas and provide Status as expression **upper(vars.newStatus)** and Correlation Key Filter as expression **vars.corKey** and provide **body** parameter as expression as follows:

```
%dw 2.0
output application/json
---
{
  "NewStatus": vars.fetchedStatus
}
```

- This operation updates the status for all messages that meet the filter parameters.
- Add **Logger** to log response and set message as expression **payload**
- Create a new sub flow **external-sys-goods-receipt-note-confirmation-flow**.
- Add **Set Variable** and set variable name as **correlationKey** and value as **\${correlation.key}** key from **mule-application.properties** file and display name as **Correlation Key**.
- Add **Set Variable** and set variable name as **searchStatus** and value as **\${search.status}** key from **mule-application.properties** file. and display name as **Search Status**

- Add **Search External System Goods Receipt Notes** Medius operation from **Mule Palette** to canvas and provide **status** as expression **upper(vars.searchStatus)** and **Correlation Key Filter** as expression **vars.correlationKey**. This operation returns fetched message status the parameters fetched in. The mandatory parameter is the status parameter, besides that the other parameters are fetched in as query string parameters to narrow down the result.
- Add **Logger** to log searched records and set message as expression **payload**
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if payload has any records.

```
sizeOf(payload) > 0
```

- Add **Set Variable** and set variable name as **pos** and value as **\${message.element.position}** key from **mule-application.properties** file and display name as **Element Position**. It will say which record user wants to process further based on position. For the first record its value will be 0, for the second record it will have 1 and so on.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if selected position is greater than or equal to zero and size of payload has record for that particular position.

```
vars.pos >= 0 and sizeOf(payload) > vars.pos
```

- Add **Set Variable** and set variable name as **elementValue** and value as expression **output application/json --- payload[vars.pos as Number]** and display name as **Get and Set Element**. It will fetch the element from a particular position.
- Add **Set Variable** and set variable name as **link** and value as expression **vars.elementValue.url splitBy ("/")** and display name as **Message Status Link Reference Value Split By /**. It will be useful to set a message Id.
- Add **Set Variable** and set variable name as **messageId** and value as expression **vars.link[-1]** and display name as **Message Id**. Here -1 means the last element from the array and it is messageId.
- Add **Set Variable** and set variable name as **messageStatus** and value as expression **vars.link[-2]** and display name as **Message Status**. Here -2 means the second last element from the array and it is messageStatus.
- Search **Get Goods Receipt Note Message Content** from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId** and Message Status as expression **vars.messageStatus**. This operation returns the status good note receipt content from a message that matches the provided status

and id. If the message id is in another status, nothing is returned.

- Add **Logger** to log response and set message as expression **payload**
- Add **Set Variable** and set variable name as **messageContent** and value as expression **payload** and display name as **Message Content**.
- Add **Flow Reference** to call another sub flow '**send-goods-receipt-note-to-ERP-and-confirmation-flow**' to process the fetched record to send it to ERP and then do confirmation on Medius by calling operation.
- Create **Sub Flow** named **send-goods-receipt-note-to-ERP-and-confirmation-flow**
- Add **try** block by dragging from **Mule Palette** and give name **Try - Block for Sending Goods Receipt Note to ERP** and keep settings default one . This block will be used for sending records to the ERP system. Right now it is not having anything except setting variables to decide which flow we need to execute further. This block can be revised as per requirement.
- Add **Set Variable** and set variable name as **isSendToERPSuccessfull** and value as expression **output application/json --- true as Boolean** and display name as **Success Flag**. Right now it is hard-coded but it can be set based on response got from send to ERP flow.
- In the Error Handling block, add **On Error Propagate** from **Mule Palette** and provide type as **ANY**
- In **On Error Propagate** block, add **Logger** to log error and set display name as **Error** and message as expression **error**
- Add **Transform Message** from **Mule Palette** and add the following in that.

```
%dw 2.0
output application/java
---
write(error, "application/json")
```

- Add **Set Variable** and set display name as **Error**, variable name as **errorRes** and value as expression **payload**.
- Add **Logger** to log error response variable and set display name as **Error Message** and message as expression **vars.errorRes**
- Add **Post A Procurement Error** Medius operation from **Mule Palette** and provide **body** as follows:

```
%dw 2.0
output application/json
---
{
  "Errors": [
    {
```

```

    "Message": vars.errorRes
  }
],
  "CorrelationKey": vars.correlationKey
}

```

- This method accepts an error message object, including a list of error messages. The error message object includes a CorrelationKey that needs to match the corresponding purchase order message.
- Add **Logger** to log response attributes and set display name as **Response Headers** message as expression **attributes**.
- After the **Try** block, add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if sending to ERP is successful or not.

```
vars.isSendToERPSuccessfull == true
```

- Add **Logger** to log success message and set message **Send Goods Receipt Note to ERP is successfully done**.
- Add **Transform Message** from **Mule Palette** and add the following in that.

```

%dw 2.0
output application/json
---
{
  OrderNumber: vars.messageContent.orderNumber,
  ExternalOrderNumber: vars.messageContent.externalOrderNumber,
  ReferenceNumber: vars.messageContent.goodsReceiptNote.referenceNumber,
  Lines: vars.messageContent.goodsReceiptNote.lines map {
    DeliveryNumber: $.deliveryNumber
  }
}

```

- Add **Logger** to log response attributes and set the message as expression **payload**.
- Add **Purchase Order To Erp Or Goods Receipt Note Confirmation** Medius operation from **Mule Palette** and set body parameter as follows:

```
%dw 2.0
```

output application/json

---

```
{
  "CorrelationKey": vars.correlationKey,
  "PurchaseOrderConfirmation": {
    "OrderNumber": payload.OrderNumber,
    "ExternalOrderNumber": payload.ExternalOrderNumber,
  },
  "GoodsReceiptNoteConfirmation": {
    "ReferenceNumber": payload.ReferenceNumber,
    "Lines": payload.Lines
  }
}
```

- This operation accepts a purchase-order-to-ERP confirmation with or without the goods-receipt-note part as a response to either the purchase-order-to-ERP or the goods-receipt-note confirmation or cancellation message, that was retrieved in the earlier step. The confirmation object includes a CorrelationKey that needs to match the corresponding purchase-order-to-ERP or goods-receipt-note message.
- Add **Logger** to log response attributes and set message as expression **attributes**
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check the status code of previous confirmation operation from Medius.

```
attributes.statusCode == 201
```

- Add **Logger** to log success message and set display name as **Confirmation Success** and message as **Purchase order to ERP or Goods Receipt Note Confirmation is done successfully.**
- Add **Try** block with display name as **Try - Block to Raise Error and Log in Medius** if **vars.isSendToERPSuccessfull == false** means in default block of that **Choice** block.
- Add **Raise Error** from **Mule Palette** in **Try** block and provide type as **ANY** and description as **Send Goods Receipt Note to ERP is failed.**
- In the Error Handling block, add **On Error Propagate** from **Mule Palette** and provide type as **ANY**
- In **On Error Propagate** block, add **Logger** to log error and set display name as **Error** and message as expression **error**
- Add **Set Variable** and set display name as **Error**, variable name as **errorRes** and value as expression **error.detailedDescription.**

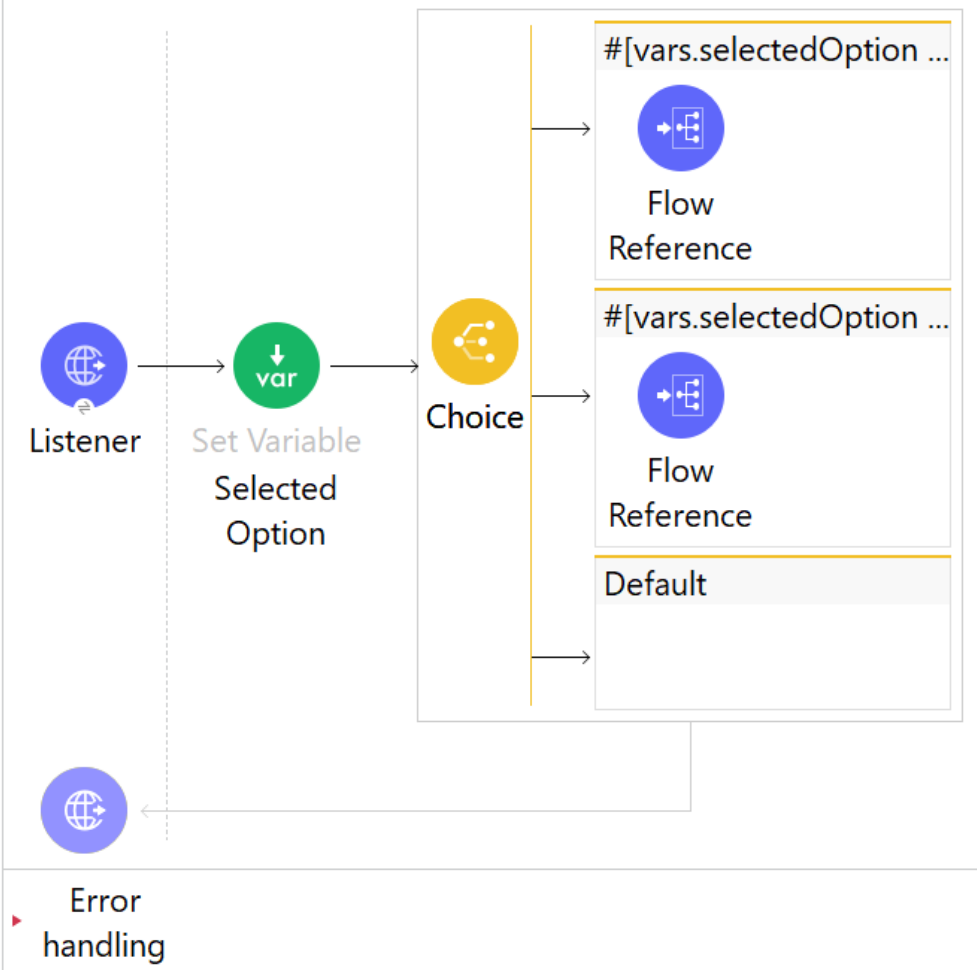
- Add **Logger** to log error response variable and set display name as **Error Message** and message as expression **vars.errorRes**
- Add **Post A Procurement Error** Medius operation from **Mule Palette** and provide **body** as follows:

```
%dw 2.0
output application/json
---
{
  "Errors": [
    {
      "Message": vars.errorRes
    }
  ],
  "CorrelationKey": vars.correlationKey
}
```

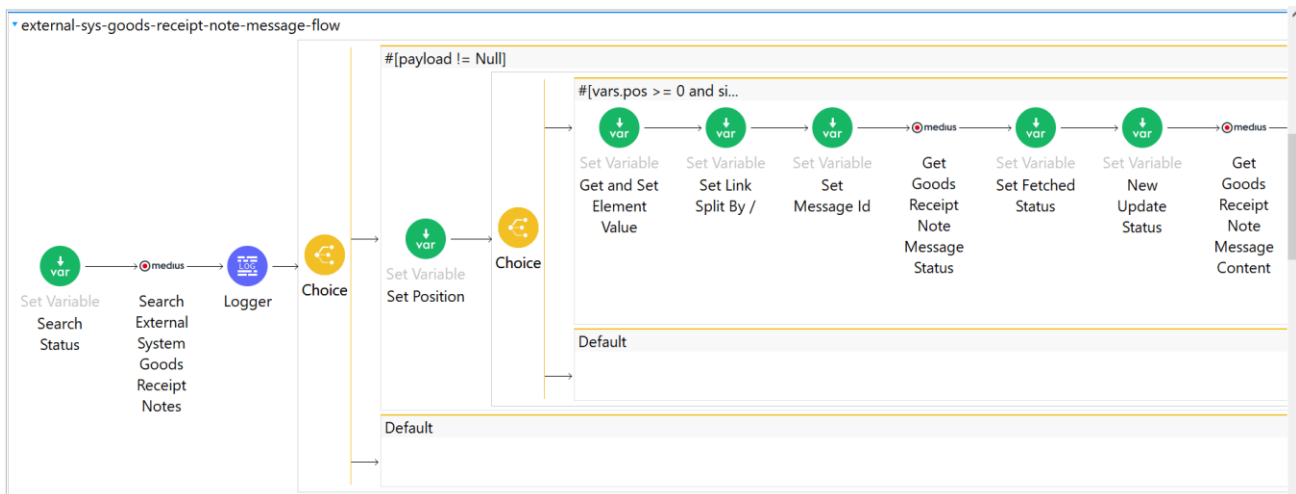
- This method accepts an error message object, including a list of error messages. The error message object includes a CorrelationKey that needs to match the corresponding purchase order message.
- Add **Logger** to log response attributes and set display name as **Response Headers** message as expression **attributes**
- After creating the flow, right-click the project and click **Run As > Mule Application**.
- Once the application is deployed, use **HTTP Listener's Path** to execute the Mule Flow.

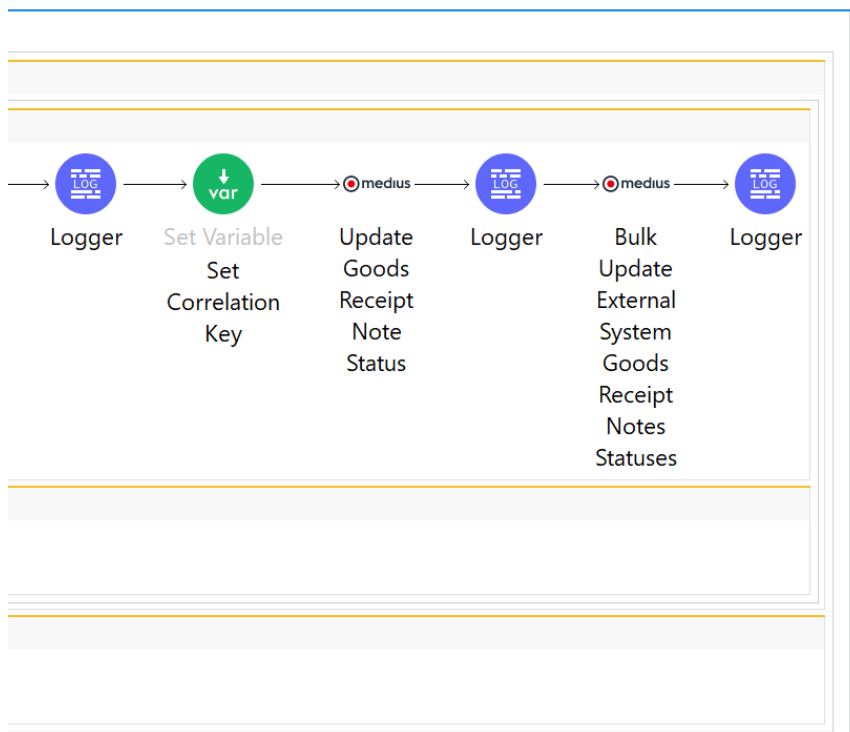
**mule-medijs-connector-external-sys-goods-receipt-note-demo**

## ▼ mule-medijs-connector-external-sys-goods-receipt-note-demo

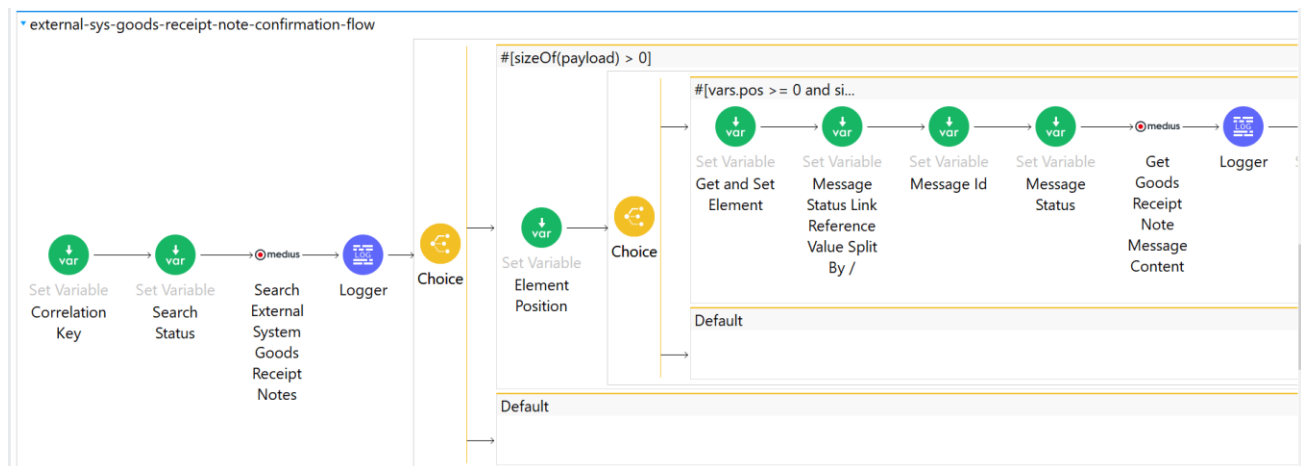


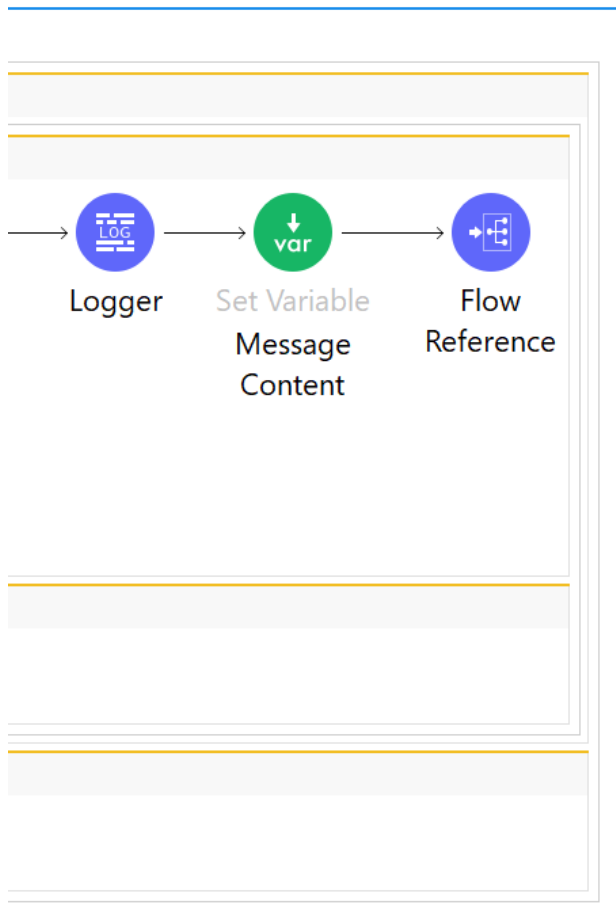
## external-sys-goods-receipt-note-message-flow



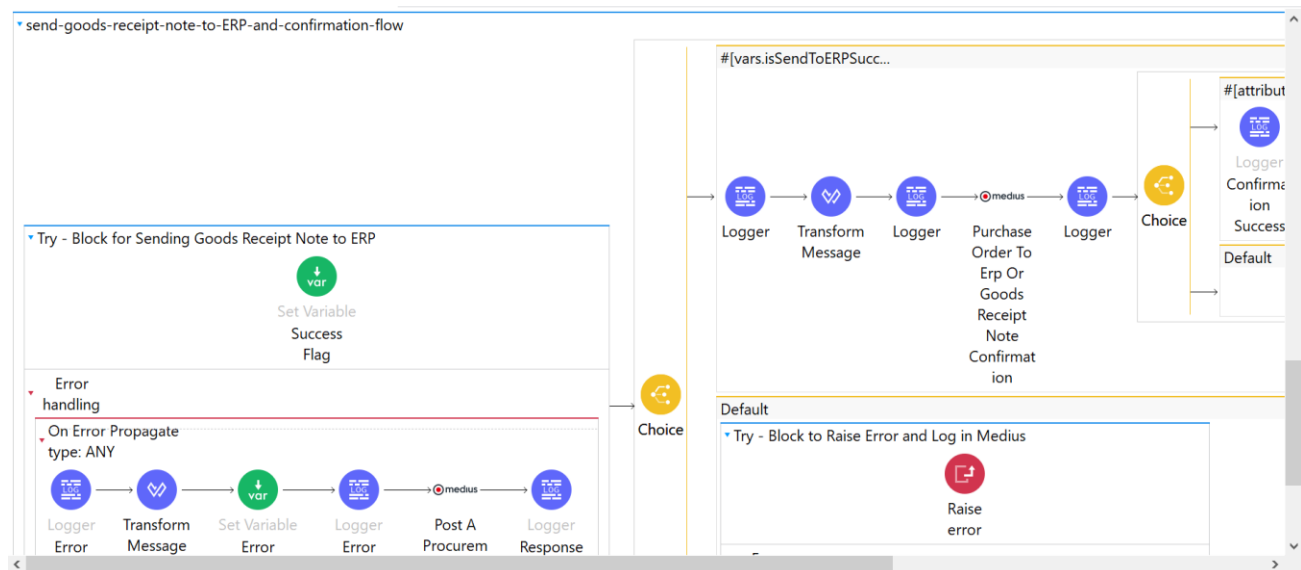


## external-sys-goods-receipt-note-confirmation-flow





## send-goods-receipt-note-to-ERP-and-confirmation-flow



▼ Try - Block for Sending Goods Receipt Note to ERP



Set Variable  
Success  
Flag

▼ Error handling

▼ On Error Propagate  
type: ANY



Logger  
Error



Transform  
Message



Set Variable  
Error



Logger  
Error  
Message



Post A  
Procurem  
ent Error



Logger  
Response  
Headers



Choice

#[vars.isSendToERPSucc...



Logger



Transform  
Message



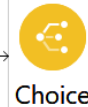
Logger



Purchase  
Order To  
Erp Or  
Goods  
Receipt  
Note  
Confirmat  
ion



Logger



Choice

#[attributes.statusCod...



Logger  
Confirmat  
ion  
Success

Default



Choice

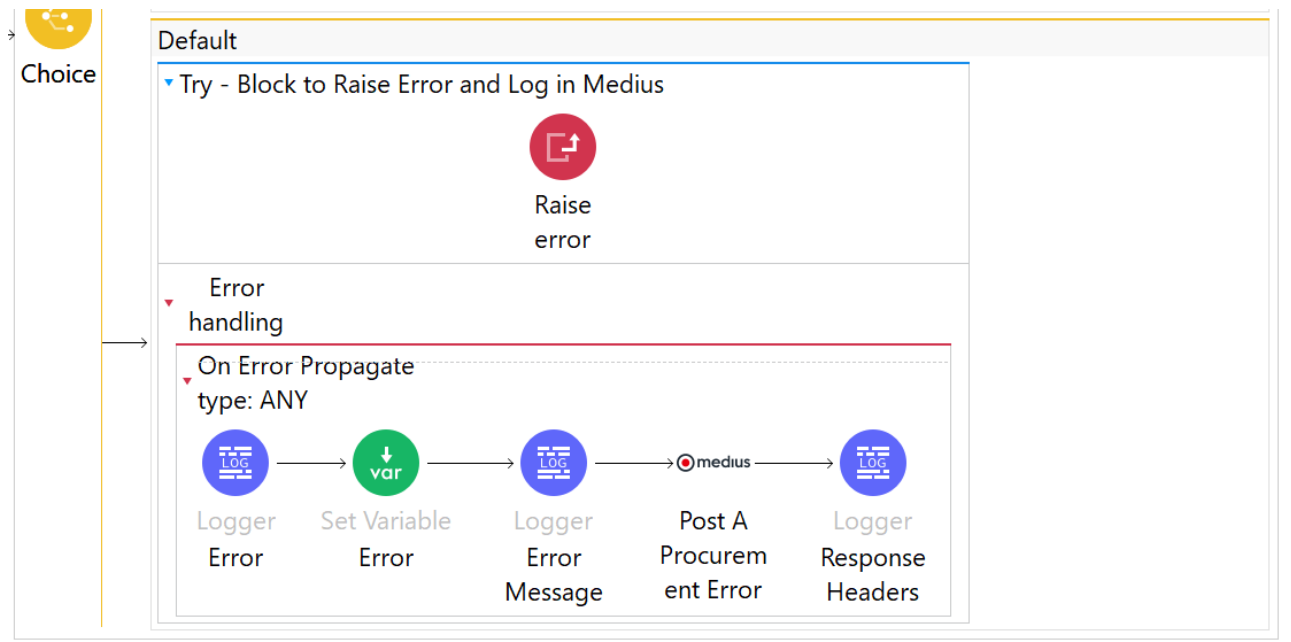
Default

▼ Try - Block to Raise Error and Log in Medius



Raise  
error

▼ Error handling



**Use Case Configuration XML :** Paste this XML code into Anypoint Studio to experiment with the flow described above.

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

<mule xmlns:ee="http://www.mulesoft.org/schema/mule/ee/core"
      xmlns:mule4-medijs-connector="http://www.mulesoft.org/schema/mule/mule4-medijs-connector"
      xmlns:http="http://www.mulesoft.org/schema/mule/http"
      xmlns="http://www.mulesoft.org/schema/mule/core"
      xmlns:doc="http://www.mulesoft.org/schema/mule/documentation"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="
http://www.mulesoft.org/schema/mule/ee/core
http://www.mulesoft.org/schema/mule/ee/core/current/mule-ee.xsd
http://www.mulesoft.org/schema/mule/core
http://www.mulesoft.org/schema/mule/core/current/mule.xsd
http://www.mulesoft.org/schema/mule/http
http://www.mulesoft.org/schema/mule/http/current/mule-http.xsd
http://www.mulesoft.org/schema/mule/mule4-medijs-connector
http://www.mulesoft.org/schema/mule/mule4-medijs-connector/current/mule-mule4-medijs-connector.xsd">

  <configuration-properties
    file="mule-application.properties" />

  <http:listener-config name="HTTP_Listener_config"
    doc:name="HTTP Listener config"
    doc:id="09f5cca2-7cd3-437e-a268-5ee0c7824a3d">
```

```

        <http:listener-connection host="0.0.0.0"
            port="8081" />
    </http:listener-config>

    <mule4-medijs-connector:config
        name="Medius_Connector_Mule4_Config"
        doc:name="Medius Connector Mule4 Config"
        doc:id="1adba4ea-02c5-4d97-a897-a0b14afe87be">
        <mule4-medijs-connector:medijsflowcloudatauth-connection
            baseUri="{medijs.base.url}">
            <mule4-medijs-connector:oauth-client-credentials
                clientId="{medijs.client.id}" tokenUrl="{medijs.token.url}"
                clientSecret="{medijs.client.secret}" scopes="{medijs.scopes}"
            />
        </mule4-medijs-connector:medijsflowcloudatauth-connection>
    </mule4-medijs-connector:config>
    <flow name="mule-medijs-connector-external-sys-goods-receipt-note-demo"
        doc:id="e8fbed08-9297-47e8-9ec1-e5b80a3ddbee">
        <http:listener doc:name="Listener"
            doc:id="cbb92494-e6d3-4425-9958-77d7957bd8f7"
            config-ref="HTTP_Listener_config"
            path="/external-sys-goods-receipt-notes" />
        <set-variable value="{selected.option}"
            doc:name="Selected Option"
            doc:id="70a16b46-f04e-4e9f-86f6-7fdc808a2f62"
            variableName="selectedOption" />
        <choice doc:name="Choice"
            doc:id="c1fa0743-3ad4-45f9-bbec-0e7f8a7a8adf">
            <when
                expression='#[vars.selectedOption ==
"MSG_SEARCH_FETCH_UPDATE"]'>
                <flow-ref doc:name="Flow Reference"
                    doc:id="506aea22-7f1a-44e6-bbe9-a7d80dec68bc"
                    name="external-sys-goods-receipt-note-message-flow"
                />
            </when>
            <when expression='#[vars.selectedOption == "CONFIRMATION"]'>
                <flow-ref doc:name="Flow Reference"
                    doc:id="7348442f-eeab-4831-8959-3b60f1c3b4f0"
                    name="external-sys-goods-receipt-note-confirmation-
flow" />
            </when>
        </choice>
    </flow>
    <sub-flow name="external-sys-goods-receipt-note-message-flow"
        doc:id="637f2827-5386-4353-897c-6c52d35626d3">
        <set-variable value="{search.status}"
            doc:name="Search Status"
            doc:id="d7ebe10e-8428-4ee8-a7fe-b516cba97a56"
            variableName="searchStatus" />
    </sub-flow>

```

```

<mule4-medijs-connector:search-external-system-goods-receipt-notes
  statusUriParam="#[upper(vars.searchStatus)]"
  doc:name="Search External System Goods Receipt Notes"
  doc:id="88a471b1-870b-4e29-aad3-6f48ec2ae219"
  config-ref="Medius_Connector_Mule4_Config" />
<logger level="INFO" doc:name="Logger"
  doc:id="9000d584-fc88-452a-ae9a-37c14be9cfaa" message="#[payload]" />
<choice doc:name="Choice" doc:id="ac79b0bf-5236-4574-b0d3-33b750b129fe" >
  <when expression="#[payload != Null]">
    <set-variable value="$[message.element.position]" doc:name="Set
Position" doc:id="d050db27-c259-45f4-b824-a3b2e9268c88" variableName="pos"/>
    <choice doc:name="Choice" doc:id="ac1c629a-143e-406b-a975-
61c252df954c">
      <when expression="#[vars.pos >= 0 and sizeOf(payload) >
vars.pos]">
        <set-variable value="#[output application/json
--- payload[vars.pos as Number]]" doc:name="Get and Set Element Value" doc:id="d7ec99c1-682a-442a-
8cd2-bc3ee7a1847a" variableName="elementValue" />
        <set-variable value="#[vars.elementValue.url
splitBy ("/")] doc:name="Set Link Split By /" doc:id="1be39b58-f9f4-41b1-a2ed-0574996f34cc"
variableName="link"/>
        <set-variable value="#[vars.link[-1]]"
doc:id="905eeb1d-8d11-487b-8817-a13cbcd1b4f1"
doc:name="Set Message Id" variableName="messageId"/>
        <mule4-medijs-connector:get-goods-receipt-
note-message-status doc:name="Get Goods Receipt Note Message Status" doc:id="5f6becb2-81ff-417d-
b234-bed76e96affc"
        config-ref="Medius_Connector_Mule4_Config"
        messageIdUriParam="#[vars.messageId]"/>
        <set-variable value="#[upper(payload.status)]"
doc:id="7e5394d2-1378-4b8e-a6d9-0d0d179a2fa8"
doc:name="Set Fetched Status" variableName="fetcheStatus" />
        <set-variable value="$[new.update.status]"
doc:id="321878c1-2ddc-4387-a9c3-6a5c893b92da"
doc:name="New Update Status" variableName="newStatus" />
        <mule4-medijs-connector:get-goods-receipt-
note-message-content doc:name="Get Goods Receipt Note Message Content" doc:id="6aac907b-ce0f-4094-
878d-5880890e5a7d"
        config-ref="Medius_Connector_Mule4_Config"
        messageStatusUriParam="#[upper(vars.fetcheStatus)]" messageIdUriParam="#[vars.messageId]"/>
        <logger level="INFO" doc:name="Logger"
doc:id="b572909d-ab44-47fe-84d4-166952e5d384" message="#[payload]" />
        <set-variable
value="#[payload.CorrelationKey]" doc:name="Set CorrelationKey" doc:id="0d015097-4d35-4561-b61b-
f2da9af021fa" variableName="corKey" />
        <mule4-medijs-connector:update-goods-
receipt-note-status doc:name="Update Goods Receipt Note Status" doc:id="3c3fbe55-994b-40bd-8178-
ea39a29605cf" config-ref="Medius_Connector_Mule4_Config"
        messageIdUriParam="#[vars.messageId]"
        messageStatusUriParam="#[vars.fetcheStatus]">
          <mule4-medijs-connector:update-
goods-receipt-note-status-body ><![CDATA#[%dw 2.0
output application/json

```

```

---
{
  "NewStatus": vars.newStatus
}}]]></mule4-medijs-connector:update-goods-receipt-note-status-body>
</mule4-medijs-connector:update-goods-
receipt-note-status>
<logger level="INFO" doc:name="Logger"
doc:id="17c8a453-1c55-49b8-9ba8-5161d7f11eba" message="#[payload]"/>
<mule4-medijs-connector:bulk-update-
external-system-goods-receipt-notes-statuses doc:name="Bulk Update External System Goods Receipt Notes
Statuses" doc:id="77817352-617e-4902-8ef6-a538c31e1e45" config-
ref="Medius_Connector_Mule4_Config" statusUriParam="#[upper(vars.newStatus)]"
correlationKeyFilterQueryParam="#[vars.corKey]">
<mule4-medijs-connector:bulk-
update-external-system-goods-receipt-notes-statuses-body><![CDATA[#[%dw 2.0
output application/json
---
{
  "NewStatus": vars.fetchedStatus
}}]]></mule4-medijs-connector:bulk-update-external-system-goods-receipt-notes-statuses-body>
</mule4-medijs-connector:bulk-update-
external-system-goods-receipt-notes-statuses>
<logger level="INFO" doc:name="Logger"
doc:id="9cbc10ef-cfae-4caa-91ef-2ca9fde8cd5b" message="#[payload]"/>
</when>
</choice>
</when>
</choice>
</sub-flow>
<sub-flow name="external-sys-goods-receipt-note-confirmation-flow"
doc:id="933b5d9f-1b38-4df2-8a2e-a4a2c42be1fa">
<set-variable value="{correlation.key}"
doc:name="Correlation Key"
doc:id="d2086eef-1b12-421b-afae-
081fb9c4706f"
variableName="correlationKey" />
<set-variable value="{search.status}" doc:name="Search Status" doc:id="3cd039ad-
5daa-444a-af27-a061073c8291" variableName="searchStatus" />
<mule4-medijs-connector:search-external-system-goods-receipt-notes
doc:name="Search External System Goods Receipt Notes" doc:id="95ebe3d2-5732-4c8f-8427-
53cba5829ca0" config-ref="Medius_Connector_Mule4_Config"
statusUriParam="#[upper(vars.searchStatus)]" correlationKeyFilterQueryParam="#[vars.correlationKey]"/>
<logger level="INFO" doc:name="Logger" doc:id="fe7d85b6-e326-43c1-8c52-
87ce1ba1898d" message="#[payload]"/>
<choice doc:name="Choice"
doc:id="c0c1d1f8-1941-4c9c-a40a-15a741854152">
<when expression="#[sizeOf(payload) > 0]">
<set-variable value="{message.element.position}"

```

```

doc:name="Element Position"
doc:id="caa6fbfd-5350-472d-8dca-8533e5611cf0"

variableName="pos" />

<choice doc:name="Choice" doc:id="635a0218-89f7-4217-8095-
19596c130aa6">

    <when expression="#[vars.pos >= 0 and
sizeOf(payload) > vars.pos]">

        <set-variable value="#[output application/json
--- payload[vars.pos as Number]]" doc:name="Get and Set Element" doc:id="87e552b7-4baa-4538-a36d-
b1fbd92e99c6" variableName="elementValue" />

        <set-variable value="#[vars.elementValue.url
splitBy ("/")]'" doc:name="Message Status Link Reference Value Split By "/" doc:id="e9f660b9-e44b-4327-af5b-
f72c53779fa9" variableName="link" />

        <set-variable value="#[vars.link[-1]]"
doc:name="Message Id" doc:id="84299277-dcde-4141-82d1-24c8598b8662" variableName="messageId" />

        <set-variable value="#[vars.link[-2]]"
doc:name="Message Status" doc:id="0be191fd-df02-4746-99f1-a2e6dbeea64e"
variableName="messageStatus" />

        <mule4-medijs-connector:get-goods-receipt-
note-message-content doc:name="Get Goods Receipt Note Message Content" doc:id="24e7290f-242f-41f9-
9d52-c7e8a92fc1d1"
config-ref="Medius_Connector_Mule4_Config"
messageStatusUriParam="#[vars.messageStatus]" messageIdUriParam="#[vars.messageId]" />

        <logger level="INFO" doc:name="Logger"
doc:id="39d5a947-d712-4dcb-9053-2a41c102c0da" message="#[payload]" />

        <set-variable value="#[payload]"
doc:name="Message Content" doc:id="3f1ea41c-2ead-41b9-bd34-f13d28780cd1"
variableName="messageContent" />

        <flow-ref doc:name="Flow Reference"
name="send-goods-receipt-note-to-ERP-and-
confirmation-flow" />

    </when>

</choice>

</when>

</choice>

</sub-flow>

<sub-flow name="send-goods-receipt-note-to-ERP-and-confirmation-flow" doc:id="f2a6e7bb-f9f0-
4e7b-a215-290733bb9172" >

    <try doc:name="Try - Block for Sending Goods Receipt Note to ERP ">

        <set-variable value="#[output application/json --- true as Boolean]" doc:name="Success
Flag" doc:id="5d7ca41c-52bb-4b28-8db9-890a6ee14c56" variableName="isSendToERPSuccessful"/>

        <error-handler >

            <on-error-propagate enableNotifications="true"
logException="true" doc:name="On Error Propagate" doc:id="6631e378-ffbe-4f88-98fb-db72e61bdc23"
type="ANY">

                <logger level="INFO" doc:name="Error"
doc:id="d6a5e9a2-fd07-40de-9a29-b5650584f497" message="#[error]" />

                <ee:transform doc:name="Transform Message"
doc:id="680890ef-af82-4c3b-a073-f40fb3060465" >

                    <ee:message >

                        <ee:set-payload ><![CDATA[%dw 2.0

```

output application/java

---

write(error, "application/json"))]></ee:set-payload>

</ee:message>

</ee:transform>

<set-variable value="#[payload]" doc:name="Error"  
doc:id="3cbaf504-498e-448c-8d93-f83d47a3eb93" variableName="errorRes"/>

<logger level="INFO" doc:name="Error Message"  
doc:id="ca83044d-df27-4565-91b2-071a676d7ac0" message="#[vars.errorRes]" />

<mule4-medijs-connector:post-a-procurement-error  
doc:name="Post A Procurement Error" doc:id="76afe940-e74f-4fff-a5f8-0abf7c36753d" config-  
ref="Medius\_Connector\_Mule4\_Config" >

<mule4-medijs-connector:post-a-

procurement-error-body ><![CDATA[#[%dw 2.0  
output application/json

---

```
{  
  "Errors": [  
    {  
      "Message": vars.errorRes  
    }  
  ],  
  "CorrelationKey": vars.correlationKey  
}]></mule4-medijs-connector:post-a-procurement-error-body>
```

</mule4-medijs-connector:post-a-procurement-error>

<logger level="INFO" doc:name="Response Headers"  
doc:id="5b5723f0-894e-4caa-aab7-231b561b1039" message="#[attributes]" />

</on-error-propagate>

</error-handler>

</try>

<choice doc:name="Choice" doc:id="5e5dea60-1ec2-4290-a263-62581ef58b45" >

<when expression="#[vars.isSendToERPSuccessfull == true]">

<logger level="INFO" doc:name="Logger" doc:id="83e8906b-64f1-  
4daa-8f45-ed8c307a8442" message="Send Goods Receipt Note to ERP is successfully done." />

<ee:transform doc:name="Transform Message" doc:id="46a85af1-  
cf07-4ffe-bea0-ca21fa832d56">

<ee:message>

<ee:set-payload><![CDATA[%dw 2.0

output application/json

---

```
{  
  OrderNumber: vars.messageContent.orderNumber,  
  ExternalOrderNumber: vars.messageContent.externalOrderNumber,  
  ReferenceNumber: vars.messageContent.goodsReceiptNote.referenceNumber,  
  Lines: vars.messageContent.goodsReceiptNote.lines map {  
    DeliveryNumber: $.deliveryNumber  
  }  
}
```

```

]]]></ee:set-payload>

                                </ee:message>
                                </ee:transform>
                                <logger level="INFO" doc:name="Logger" doc:id="04c42044-9fd4-
47f9-be59-d9d0b80bac4a" message="#[payload]" />
                                <mule4-mediustConnector:purchase-order-to-erp-or-goods-
receipt-note-confirmation doc:name="Purchase Order To Erp Or Goods Receipt Note Confirmation"
doc:id="22c6ff00-6beb-4fe6-a226-83c7320ee012" config-ref="Medius_Connector_Mule4_Config" >
                                <mule4-mediustConnector:purchase-order-to-erp-or-
goods-receipt-note-confirmation-body ><![CDATA[#[%dw 2.0
output application/json
---
{
  "CorrelationKey": vars.correlationKey,
  "PurchaseOrderConfirmation": {
    "OrderNumber": payload.OrderNumber,
    "ExternalOrderNumber": payload.ExternalOrderNumber,
  },
  "GoodsReceiptNoteConfirmation": {
    "ReferenceNumber": payload.ReferenceNumber,
    "Lines": payload.Lines
  }
}]]]></mule4-mediustConnector:purchase-order-to-erp-or-goods-receipt-note-confirmation-body>
                                </mule4-mediustConnector:purchase-order-to-erp-or-goods-
receipt-note-confirmation>
                                <logger level="INFO" doc:name="Logger" doc:id="bec2cef6-9548-430e-86f1-
173330a60317" message="#[attributes]" />

                                <choice doc:name="Choice"
doc:id="b0dc4440-4aaa-40d9-a185-e0fc564653df">
                                <when
                                  expression="#[attributes.statusCode == 201]">
                                  <logger level="INFO" doc:name="Confirmation Success"
doc:id="2b6b1065-7d66-4d3f-8f72-399940e1fb49" message="Purchase order to ERP or Goods Receipt Note
Confirmation is done successfully." />
                                  </when>
                                </choice>
                                </when>
                                <otherwise >
                                  <try doc:name="Try - Block to Raise Error and Log in Medius"
doc:id="e0af452d-eb7d-41cb-bc44-1b5a72b302b9" >
                                  <raise-error doc:name="Raise error" doc:id="a709b950-
d9cb-493f-b4ea-8bb7e3b2c7d3" type="ANY" description="Send Goods Receipt Note to ERP is failed." />
                                  <error-handler >
                                    <on-error-propagate
enableNotifications="true" logException="true" doc:name="On Error Propagate" doc:id="2ec992f1-b42d-
4d16-9148-391a2bc6c662" type="ANY">
                                      <logger level="INFO"
doc:name="Error" doc:id="81d6be9f-c5ce-4e49-b960-3fcc6a209b15" message="#[error]" />

```

```

                                <set-variable          value="#[error.detailedDescription]"
doc:name="Error" doc:id="e3c0ecd6-e3c1-4103-ac78-af6c77be82b0" variableName="errorRes"/>
                                <logger    level="INFO"    doc:name="Error    Message"
doc:id="891d6521-40d8-4075-9a35-b3f66f15410b" message="#[vars.errorRes]"/>
                                <mule4-medi-us-connector:post-a-
procurement-error    doc:name="Post    A    Procurement    Error"    doc:id="daa11bdd-cf1b-4dd5-b448-
f8851ad9d45f" config-ref="Medius_Connector_Mule4_Config" >
                                <mule4-medi-us-connector:post-a-
procurement-error-body ><![CDATA[#[%dw 2.0
output application/json
---
{
  "Errors": [
    {
      "Message": vars.errorRes
    }
  ],
  "CorrelationKey": vars.correlationKey
}]]></mule4-medi-us-connector:post-a-procurement-error-body>
                                </mule4-medi-us-connector:post-a-procurement-error>
                                <logger    level="INFO"    doc:name="Response    Headers"
doc:id="c45525b1-480b-4ac4-947b-cedfc078a5ae" message="#[attributes]" />
                                </on-error-propagate>
                                </error-handler>
                                </try>
                                </otherwise>
                                </choice>
                                </sub-flow>
</mule>

```

## Export of Data - Image or Attachment

This use case demonstrates the usage of Get Document Images Links, Get Document Attachments Links operation to fetch links for images and attachments based on provided document id. Further fetched link reference is used to get image or attachment based on document id and image / attachment hash value using Get Image or Get Attachment operation. Users can execute flow based on selection of flow to execute.

- In Anypoint Studio, click **File > New > Mule Project**, name the project **mule-medi-us-connector-export-data-demo** and click Finish.
- In the search field, type **Http** and drag the **HTTP Listener** to the canvas.
- Click the **HTTP Listener**, click the green plus sign to the right of Connector Configuration. On the next screen, add the host and port, click OK
- Provide **Path** to HTTP Listener (Example: **/export-data**)
- On the Mule Palette, search for **Set Variable** and drag it onto the canvas. Set value of variable from

**mule-application.properties** file key “**selected.export.option**” as **\${selected.export.option}** and give name of variable as **selectedExportOption** and display name as **Selected Export Option**.

- There are two options available for execution of two different flows. This option user can change from **mule-application.properties** file.
  - **IMAGES** - execute images related fetch link and then file flow
  - **ATTACHMENTS** - execute attachment related fetch link and then file flow
- Add **Set Variable** and drag it onto the canvas. Set value of variable from **mule-application.properties** file key “**document.id**” as **\${document.id}** and give name of variable as **documentId** and display name as **Document Id**.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- There will be two **When** blocks for added **Choice**. Those will be as follows.
- Add below condition as expression in **Choice** to check if the user wants to execute images related flow.

```
vars.selectedExportOption == 'IMAGES'
```

- Add **Get Document Images Links** Medius operation from **Mule Palette** and provide document id as expression **vars.documentId**. This operation returns a list of file links for a document identified by parameter documentId. Each file link consists of the original file name and a link for downloading this file. Draft, migrated and long-term archived documents are not supported by this API. If the user has mentioned a selected export operation as IMAGES then this function will be called in the demo for the mentioned document id in the properties file. (mentioned in **mule-application.properties** file).
- Add below condition as expression in **Choice** to check if the user wants to execute attachment related flow.

```
vars.selectedExportOption == 'ATTACHMENTS'
```

- Add **Get Document Attachments Links** Medius operation from **Mule Palette** and provide document id as expression **vars.documentId**. This operation returns the list of attachment links for a document identified by parameter documentId. Each attachment link consists of the original attachment name, and a link for downloading this attachment. Draft, migrated and long-term archived documents are not supported by this API. If the user has mentioned the selected export operation as ATTACHMENTS then this function will be called in the demo for the mentioned document id in the properties file. (mentioned in **mule-application.properties** file).
- After this **Choice** block with these two options, add **Choice** from **Mule Palette**.
- Add below condition as expression in **Choice** to check if any records are fetched or not.

```
sizeof(payload) > 0
```

- Add **Flow Reference** to call another sub flow '**export-image-or-attachments-flow**' to fetch a particular image or attachment flow.
- Create a new sub flow **export-image-or-attachments-flow**.
- Add **Set Variable** and set variable name as **elementVal** and value as expression **payload[0]** and display name as **Get and Fetch Element**. Here we are fetching the first record from payload.
- Add **Set Variable** and set variable name as **link** and value as expression **vars.elementVal.ref splitBy ("/")** and display name as **Link Reference Value Split By /**.
- Add **Set Variable** and set variable name as **hashValue** and value as expression **vars.link[-1]** and display name as **Hash Value**.
- Add **Choice** from **Mule Palette**.
- Add below condition as expression in **Choice** to check if the user wants to export image.

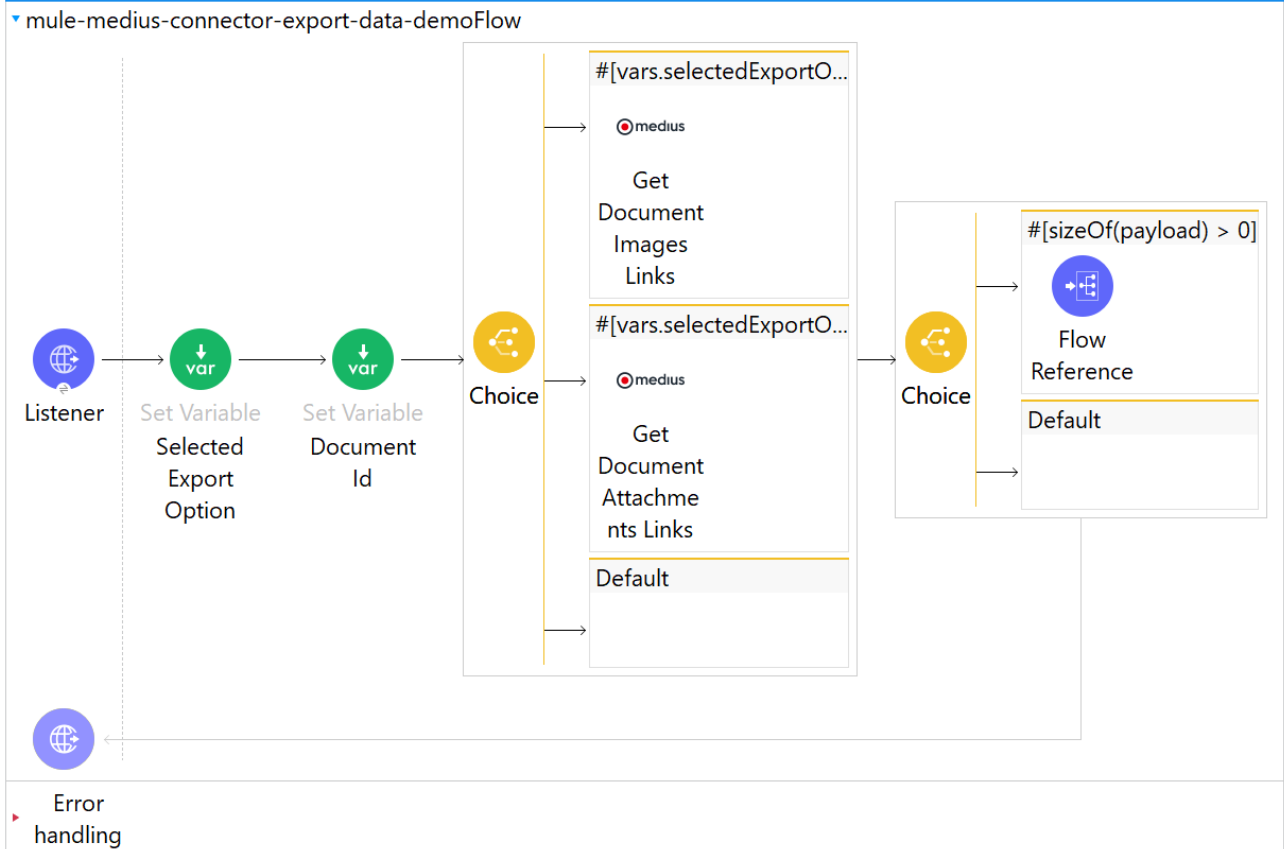
```
vars.selectedExportOption == 'IMAGES'
```

- Add below condition as expression in **Choice** to check if the user wants to export attachment.

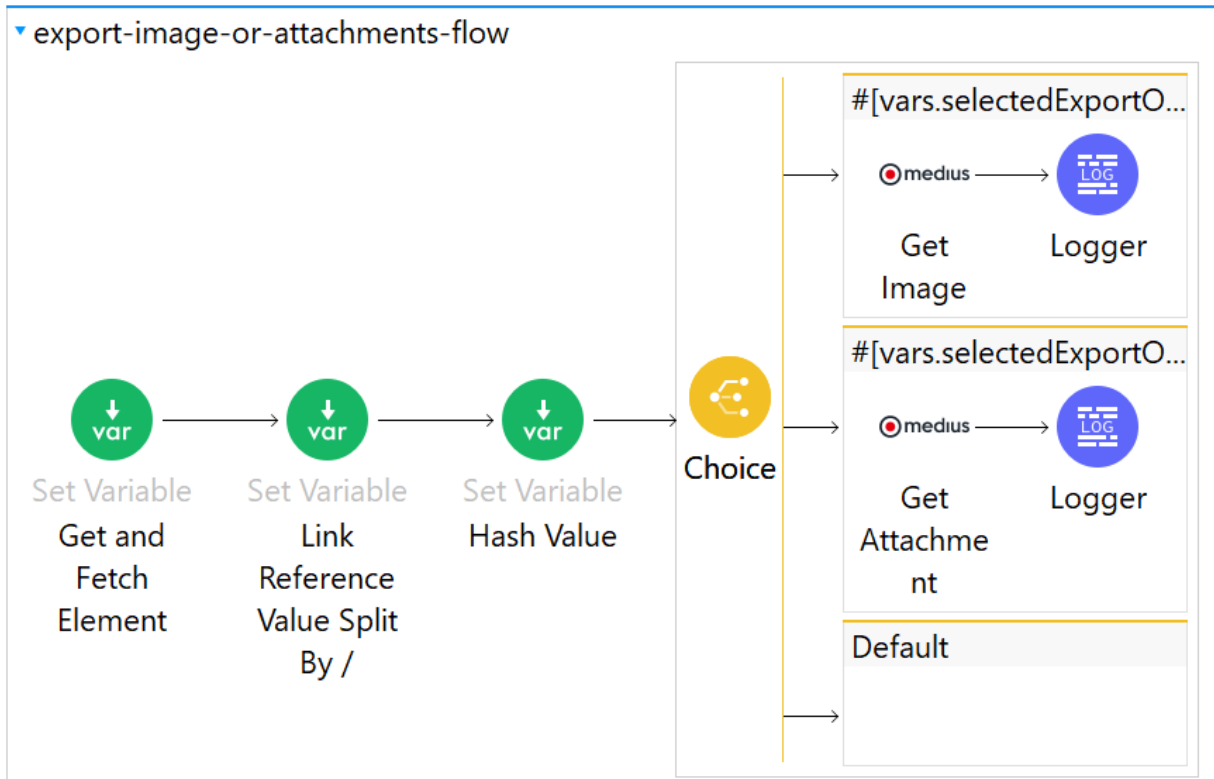
```
vars.selectedExportOption == 'ATTACHMENTS'
```

- If **vars.selectedExportOption == 'IMAGES'** then add **Get Image** Medius operation from **Mule Palette** and provide document id as expression **vars.documentId** and image hash as expression **vars.hashValue**. This operation returns a file with the given hash for a document identified by parameter documentId. Hash value will be available as part of "ref" parameter link value fetched using Get Document Images Links operation.
- Add **Logger** to log Response Headers and set the message as expression **attributes**.
- If **vars.selectedExportOption == 'ATTACHMENTS'** then add **Get Attachment** Medius operation and provide document id as expression **vars.documentId** and attachment hash as expression **vars.hashValue**. This operation returns a file with the given hash for a document identified by parameter documentId. Hash value will be available as part of "ref" parameter link value fetched using Get Document Attachment Links operation.
- Add **Logger** to log Response Headers and set the message as expression **attributes**.
- After creating the flow, right-click the project and click **Run As > Mule Application**.
- Once the application is deployed, use **HTTP Listener's Path** to execute the Mule Flow.

**mule-medijs-connector-export-data-demoFlow**



### export-image-or-attachments-flow



**Use Case Configuration XML** : Paste this XML code into Anypoint Studio to experiment with the flow described above.

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

<mule
  xmlns:mule4-medijs-connector="http://www.mulesoft.org/schema/mule/mule4-medijs-
connector"
  xmlns:http="http://www.mulesoft.org/schema/mule/http"
  xmlns="http://www.mulesoft.org/schema/mule/core"
  xmlns:doc="http://www.mulesoft.org/schema/mule/documentation"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.mulesoft.org/schema/mule/core
http://www.mulesoft.org/schema/mule/core/current/mule.xsd
http://www.mulesoft.org/schema/mule/http
http://www.mulesoft.org/schema/mule/http/current/mule-http.xsd
http://www.mulesoft.org/schema/mule/mule4-medijs-connector
http://www.mulesoft.org/schema/mule/mule4-medijs-connector/current/mule-mule4-medijs-
connector.xsd">

  <configuration-properties
    file="mule-application.properties" />

  <http:listener-config name="HTTP_Listener_config"
    doc:name="HTTP Listener config"
    doc:id="09f5cca2-7cd3-437e-a268-5ee0c7824a3d">
    <http:listener-connection host="0.0.0.0"
      port="8081" />
  </http:listener-config>

  <mule4-medijs-connector:config
    name="Medius_Connector_Mule4_Config"
    doc:name="Medius Connector Mule4 Config"
    doc:id="1adba4ea-02c5-4d97-a897-a0b14afe87be">
    <mule4-medijs-connector:medijsflowcloudauth-connection
      baseUri="{medijs.base.url}">
      <mule4-medijs-connector:oauth-client-credentials
        clientId="{medijs.client.id}" tokenUrl="{medijs.token.url}"
        clientSecret="{medijs.client.secret}" scopes="{medijs.scopes}"
      />
    </mule4-medijs-connector:medijsflowcloudauth-connection>
  </mule4-medijs-connector:config>

  <flow name="mule-medijs-connector-export-data-demoFlow"
    doc:id="44b03b34-9f23-49a9-b06c-359d8bc8bac2">
    <http:listener doc:name="Listener"
      doc:id="a25dfd77-e60b-4e52-9d74-97b820122e98">
```

```

        config-ref="HTTP_Listener_config" path="/export-data">
</http:listener>
<set-variable value="{selected.export.option}"
    doc:name="Selected Export Option"
    doc:id="71499e5f-32a7-44d4-b63a-02618b4dc779"
    variableName="selectedExportOption" />

<set-variable value="{document.id}"
    doc:name="Document Id" doc:id="f6d9b48c-4447-44d5-a617-4c52d3ec3087"
    variableName="documentId" />

<choice doc:name="Choice"
    doc:id="6410c523-7f56-4d96-b54f-d9c13a4a8951">
    <when expression="#[vars.selectedExportOption == 'IMAGES']">
        <mule4-medijs-connector:get-document-images-links
            doc:name="Get Document Images Links"
            doc:id="12e699de-06e9-4be1-adb4-9f5771cb72a0"
            config-ref="Medius_Connector_Mule4_Config"
            documentIdUriParam="#[vars.documentId]" />

    </when>
    <when expression="#[vars.selectedExportOption == 'ATTACHMENTS']">
        <mule4-medijs-connector:get-document-attachments-links
            doc:name="Get Document Attachments Links"
            doc:id="e67cfbf1-e4ec-4f10-87e1-92b635b0ddfc"
            config-ref="Medius_Connector_Mule4_Config"
            documentIdUriParam="#[vars.documentId]" />

    </when>
</choice>
<choice doc:name="Choice"
    doc:id="8c8fb826-57fb-4c7f-ab3b-0c7ac0ed544e">
    <when expression="#[sizeOf(payload) > 0]">

        <flow-ref doc:name="Flow Reference"
            doc:id="6b91a0b3-7717-44dc-9677-
eadaff02e88a"
            name="export-image-or-
attachments-flow" />

    </when>
</choice>
</flow>

<sub-flow name="export-image-or-attachments-flow"
    doc:id="34410947-da1a-4303-ad98-db247c9e4726">

    <set-variable
        value="#[payload[0]]"

```

```

doc:name="Get and Fetch Element" doc:id="f5f57c01-2f2f-404e-a34f-
1dda04f655a0"
    variableName="elementVal" />

    <set-variable value='#[vars.elementVal.ref splitBy ("/")]'
        doc:name="Link Reference Value Split By /"
        doc:id="86763b09-af3d-42c0-b9e6-e822ef37b8a0" variableName="link" />
    <set-variable value='#[vars.link[-1]]'
        doc:name="Hash Value" doc:id="b1d6806b-468d-4b1d-a750-ce93466c1747"
        variableName="hashValue" />

    <choice doc:name="Choice"
        doc:id="2d8d8b66-70c5-4fec-bc62-8e3db29163a8">
        <when expression='#[vars.selectedExportOption == 'IMAGES']">
            <mule4-medijs-connector:get-image
                doc:name="Get Image" doc:id="3ce999de-5b2b-497e-
8864-07faaf3c4088"

                documentIdUriParam="#[vars.documentId]"
                imageHashUriParam="#[vars.hashValue]"
                config-ref="Medius_Connector_Mule4_Config" />
            <logger level="INFO" doc:name="Logger" doc:id="79d88864-a679-
42d9-a76b-03b8a3bbb45f" message="#[attributes]" />
        </when>
        <when expression='#[vars.selectedExportOption == 'ATTACHMENTS']">
            <mule4-medijs-connector:get-attachment
                doc:name="Get Attachment"
                doc:id="41c3cf2a-c4a4-47c5-ad05-255192b5868c"
                config-ref="Medius_Connector_Mule4_Config"
                documentIdUriParam="#[vars.documentId]"
                attachmentHashUriParam="#[vars.hashValue]" />
            <logger level="INFO" doc:name="Logger" doc:id="51c248af-4a96-
4034-92e6-d10c0b5e44f1" message="#[attributes]" />
        </when>
    </choice>
</sub-flow>
</mule>

```

## Procurement Purchase Order Confirmation and Cancel Confirmation and Search, Fetch, Update Status

This use case demonstrates the usage of various Purchase Order operations for example Search, Fetch Message Status, Fetch Message Content, Update Specific as well as Bulk Update Messages status, Purchase Order Request Confirmation as well as Cancellation Confirmation flow based on selection of flow to execute.

- In Anypoint Studio, click **File > New > Mule Project**, name the project **mule-medijs-connector-**

**purchase-order-demo** and click Finish.

- In the search field, type **Http** and drag the **HTTP Listener** to the canvas.
- Click the **HTTP Listener**, click the green plus sign to the right of Connector Configuration. On the next screen, add the host and port, click OK
- Provide **Path** to HTTP Listener (Example: **/purchase-order**)
- On the Mule Palette, search for **Set Variable** and drag it onto the canvas. Set value of variable from **mule-application.properties** file key “**selected.option**” as **\${selected.option}** and give name of variable as **selectedOpt** and display name as **Selected Option**.
- There are three options available for execution of three different flows. This option user can change from **mule-application.properties** file.
  - **MSG\_SEARCH\_FETCH\_UPDATE** - execute Search, Fetch Status, Fetch Content, Update Status and Bulk Update Status operations related flow
  - **PURCHASE\_ORDER\_CONFIRMATION** - Procurement Purchase Order Confirmation flow.
  - **PURCHASE\_ORDER\_CANCEL\_CONFIRMATION** - Procurement Purchase Order Cancellation Confirmation flow.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- There will be three **When** blocks for added **Choice**. Those will be as follows.
- Add below condition as expression in **Choice** to check if user wants to execute Search, Fetch Status, Fetch Content, Update Status and Bulk Update Status operations related flow.

```
vars.selectedOpt == 'MSG_SEARCH_FETCH_UPDATE'
```

- Add **Flow Reference** to call another sub flow '**purchase-order-message-flow**' to Search, Fetch Status, Fetch Content, Update Status and Bulk Update Status operations related flow.
- Add below condition as expression in **Choice** to check if the user wants to execute Procurement Purchase Order Confirmation operation related flow.

```
vars.selectedOpt == 'PURCHASE_ORDER_CONFIRMATION'
```

- Add **Flow Reference** to call another sub flow '**purchase-order-request-confirmation-flow**' to execute Procurement Purchase Order Confirmation operation related flow.
- Add below condition as expression in **Choice** to check if the user wants to execute Procurement Purchase Order Cancel Confirmation operation related flow.

```
vars.selectedOpt == 'PURCHASE_ORDER_CANCEL_CONFIRMATION'
```

- Add **Flow Reference** to call another sub flow '**purchase-order-request-cancel-confirmation-flow**' to execute Procurement Purchase Order Confirmation operation related flow.
- Create a new sub flow **purchase-order-message-flow**.
- Add **Set Variable** and set variable name as **searchStatus** and value as **\${search.status}** key from **mule-application.properties** file and display name as **Search Status**.
- Add **Search Purchase Orders** Medius operation from **Mule Palette** to canvas and provide **status** as expression **upper(vars.searchStatus)**. This operation returns all messages that match the parameters sent in. The mandatory parameter is the status parameter, besides that the other parameters are sent in as query string parameters to narrow down the result.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if payload has any record.

sizeof(payload) > 0

- Add **Set Variable** and set variable name as **pos** and value as **\${message.element.position}** key from **mule-application.properties** file and display name as **Element Position**. It will say which record user wants to process further based on position. For the first record its value will be 0, for the second record it will have 1 and so on.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if selected position is greater than or equal to zero and size of payload has record for that particular position.

vars.pos >= 0 and sizeof(payload) > vars.pos

- Add **Set Variable** and set variable name as **elementValue** and value as expression **output application/json --- payload[vars.pos as Number]** and display name as **Get and Set Element**. It will fetch the element from a particular position.
- Add **Set Variable** and set variable name as **link** and value as expression **vars.elementValue.url splitBy ("/")** and display name as **Message Status Link Reference Value Split By /**. It will be useful to set a message Id and message status.
- Add **Set Variable** and set variable name as **messageId** and value as expression **vars.link[-1]** and display name as **Message Id** here -1 means last element from array and it is messageId.
- Add **Set Variable** and set variable name as **messageStatus** and value as expression **vars.link[-2]** and display name as **Message Status** here -2 means second last element from array and it is messageStatus.

- Search **Get Purchase Order Request Message Status** Medius operation from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId** . The operation with message id as input returns a status and an effective link to real purchase order content. This method allows you to check the status of a given message.
- Add **Logger** to log payload of above operation and set message as expression **payload**.
- Search **Get Purchase Order Request Message Content** from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId** and Message Status as expression **vars.messageStatus**. This operation returns the purchase order content from a message that matches the provided status and id. If the message id is in another status, nothing is returned.
- Add **Set Variable** and set variable name as **corKey** and value as expression **payload.CorrelationKey** and display name as **Correlation Key**.
- Add **Logger** to log payload of above operation and set message as expression **payload**.
- Add **Set Variable** and set variable name as **newStatus** and value as **\${new.update.status}** from mule properties file and display name as **New Update Status**. This indicates a new update status for a given record. It will be useful in further operations.
- Search **Update Purchase Order Message Status** from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId** and Message Status as expression **vars.messageStatus** and provide **body** parameter as expression as follows:

```
%dw 2.0
output application/json
---
{
    "NewStatus": vars.newStatus
}
```

- This operation updates the status of a purchase order message when processing it.
- Add **Logger** to log response and set message as expression **payload**
- Search **Bulk Update Purchase Order Statuses** from **Mule Palette** and drag to canvas and provide Status as expression **upper(vars.newStatus)** and Correlation Key Filter as expression **vars.corKey** and provide **body** parameter as expression as follows:

```
%dw 2.0
output application/json
---
{
    "NewStatus": vars.messageStatus
}
```

- This operation updates the status for all messages that meet the filter parameters.
- Add **Logger** to log response and set message as expression **payload**

- Create a new sub flow **purchase-order-request-confirmation-flow**.
- Add **Set Variable** and set variable name as **correlationkey** and value as **\${correlation.key}** key from **mule-application.properties** file and display name as **Correlation Key**.
- Add **Purchase Order Request Confirmation** Medius operation from **Mule Palette** and set **body** parameter as follows:

```
%dw 2.0
output application/json
---
{
    "CorrelationKey": vars.correlationkey,
    "ResponseData": {}
}
```

- This operation accepts a purchase order confirmation as a response to the purchase order message.
- Add **Logger** to log response attributes and set message as expression **attributes** with display name as **Response Headers**.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check the status code of previous confirmation operation from Medius.

```
attributes.statusCode == 201
```

- Add **Logger** to log success message and set message as **Purchase Order Request Confirmation is done successfully..**
- Add **Set Variable** and set display name as **Location Header Value Split By /**, variable name as **location** and value as expression **attributes.headers.location splitBy ("/")**.
- Add **Set Variable** and set display name as **Message Id**, variable name as **messageId** and value as expression **vars.location[-1]**. Here -1 means the last element from an array.
- Add **Get Purchase Order Request Message Status** Medius operation and provide message id as expression **vars.messageId**. The operation with message id as input returns a status and an effective link to real purchase order content. This method allows you to check the status of a given message.
- Add **Logger** to log response attributes and set the message as expression **attributes** with display name as **Response Headers**.
- Add **Logger** to log response payload and set message as expression **payload** with display name as **Payload**.

- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check the status code of previously added operation from Medius.

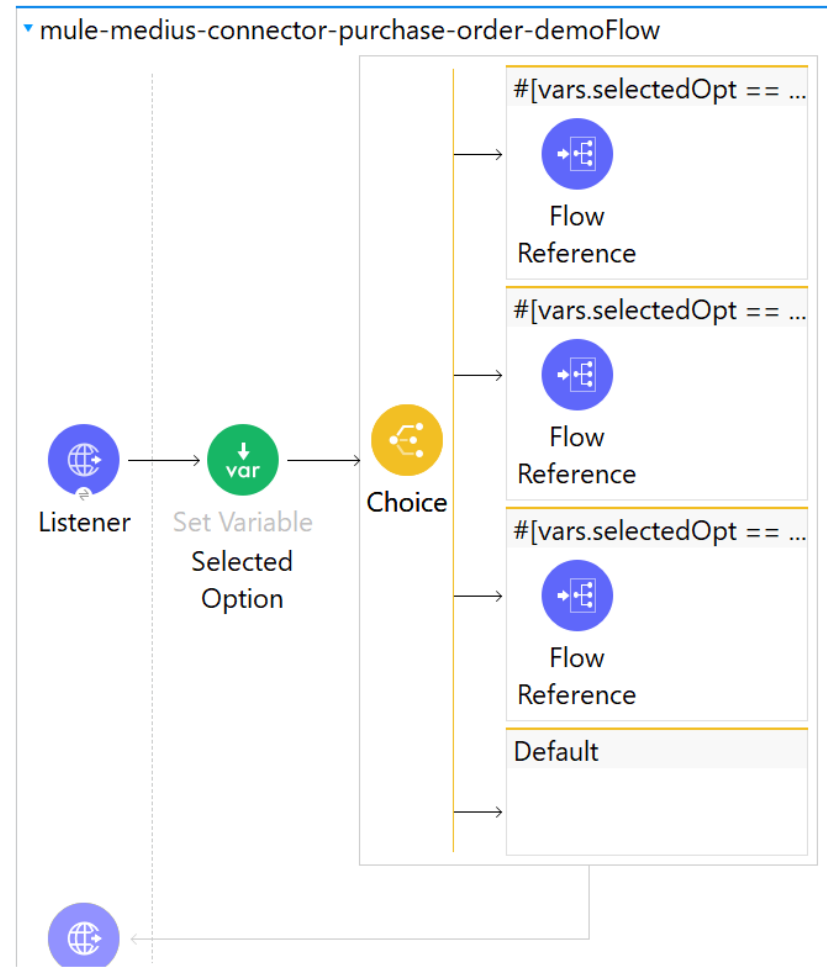
```
attributes.statusCode == 200
```

- Add **Set Variable** and set variable name as **msgStatus** and value as expression **payload.status** and display name as **Message Status**.
- Add **Get Purchase Order Request Message Content** Medius operation and provide message status as expression **vars.msgStatus** and message id as expression **vars.messageId**. This operation returns the purchase order content from a message that matches the provided status and id. If the message id is in another status, nothing is returned.
- Create a new sub flow named **purchase-order-request-cancel-confirmation-flow**.
- Add **Set Variable** and set variable name as **correlationkey** and value as **\${correlation.key}** key from **mule-application.properties** file and display name as **Correlation Key**.
- Add **Purchase Order Cancellation Confirmation** Medius operation from **Mule Palette** and set **body** parameter as follows:

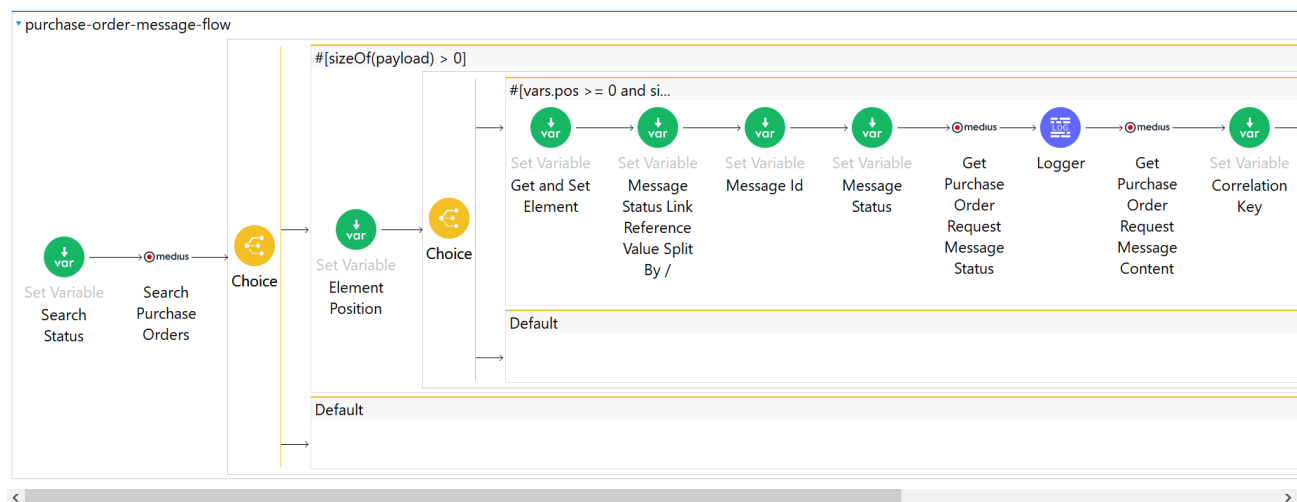
```
%dw 2.0
output application/json
---
{
  "CorrelationKey": vars.correlationkey,
  "ResponseData": {}
}
```

- This operation accepts a purchase order cancellation as a response to the purchase order message.
- Add **Logger** to log response attributes and set the message as expression **attributes** with display name as **Response Headers**.
- After creating the flow, right-click the project and click **Run As > Mule Application**.
- Once the application is deployed, use **HTTP Listener's Path** to execute the Mule Flow.

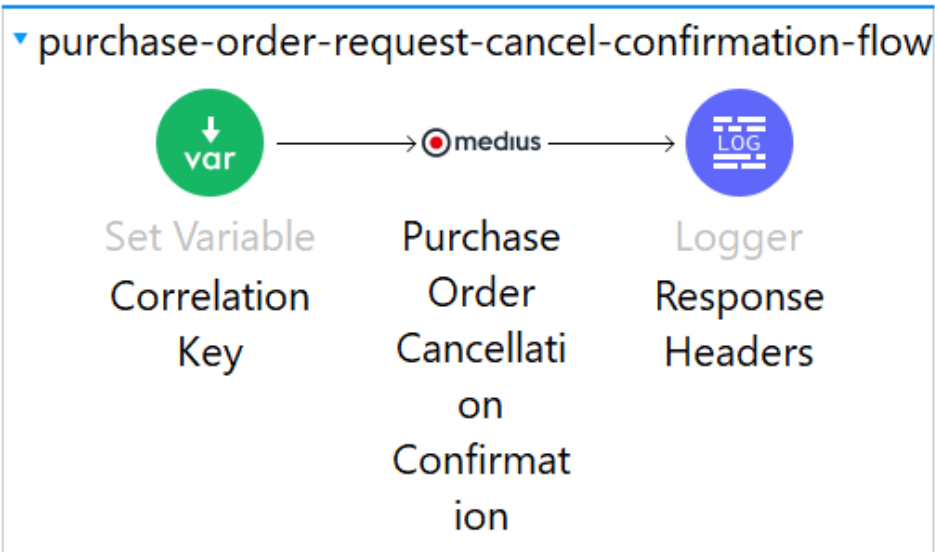
**mule-medijs-connector-purchase-order-demoFlow**



## purchase-order-message-flow







**Use Case Configuration XML** : Paste this XML code into Anypoint Studio to experiment with the flow described above.

```

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

<mule
  xmlns:mule4-medijs-connector="http://www.mulesoft.org/schema/mule/mule4-medijs-connector"
  xmlns:http="http://www.mulesoft.org/schema/mule/http"
  xmlns="http://www.mulesoft.org/schema/mule/core"
  xmlns:doc="http://www.mulesoft.org/schema/mule/documentation"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.mulesoft.org/schema/mule/core
http://www.mulesoft.org/schema/mule/core/current/mule.xsd
http://www.mulesoft.org/schema/mule/http
http://www.mulesoft.org/schema/mule/http/current/mule-http.xsd
http://www.mulesoft.org/schema/mule/mule4-medijs-connector
http://www.mulesoft.org/schema/mule/mule4-medijs-connector/current/mule-mule4-medijs-connector.xsd">

  <configuration-properties
    file="mule-application.properties" />

  <http:listener-config name="HTTP_Listener_config"
    doc:name="HTTP Listener config"
    doc:id="a2612d16-1270-488c-a6b4-6954c24d1762">
    <http:listener-connection host="0.0.0.0"
      port="8081" />
  </http:listener-config>

```

```

<mule4-mediustconnector:config
  name="Medius_Connector_Mule4_Config"
  doc:name="Medius Connector Mule4 Config"
  doc:id="f6440a0b-37db-4a30-952f-c1d306c7fd22">
    <mule4-mediustconnector:mediustflowcloudauth-connection
      baseUrl="{mediust.base.url}">
      <mule4-mediustconnector:oauth-client-credentials
        clientId="{mediust.client.id}" tokenUrl="{mediust.token.url}"
        clientSecret="{mediust.client.secret}" scopes="{mediust.scopes}"
      />
    </mule4-mediustconnector:mediustflowcloudauth-connection>
  </mule4-mediustconnector:config>

<flow name="mule-mediustconnector-purchase-order-demoFlow"
  doc:id="d1867059-04f5-4108-ba8c-c664799e705f">

  <http:listener doc:name="Listener"
    doc:id="8885df8b-423b-4a52-8a31-324c46860f8d"
    config-ref="HTTP_Listener_config" path="/purchase-order" />
  <set-variable value="{selected.option}"
    doc:name="Selected Option"
    doc:id="cf6b4f2e-4ec7-4610-9e5b-ccfb1ceba525"
    variableName="selectedOpt" />

  <choice doc:name="Choice"
    doc:id="3695ee03-17f1-4861-96c6-344499d334d1">
    <when
      expression="#[vars.selectedOpt
'MSG_SEARCH_FETCH_UPDATE']">
      <flow-ref doc:name="Flow Reference"
        doc:id="9c91580b-91db-4bfb-afb5-b48b819ab45a"
        name="purchase-order-message-flow" />
    </when>
    <when
      expression="#[vars.selectedOpt
'PURCHASE_ORDER_CONFIRMATION']">
      <flow-ref doc:name="Flow Reference"
        doc:id="28417141-9495-434a-aa6d-438447d75770"
        name="purchase-order-request-confirmation-flow" />
    </when>
    <when
      expression="#[vars.selectedOpt
'PURCHASE_ORDER_CANCEL_CONFIRMATION']">
      <flow-ref doc:name="Flow Reference"
        doc:id="6397f8b8-fac9-4b92-86c2-fc3afa0ea873"

```

```

name="purchase-order-request-cancel-confirmation-
flow" />
    </when>
    </choice>
</flow>

<sub-flow name="purchase-order-message-flow"
    doc:id="e19ae7e0-ae45-4bee-b2ec-5f5299dc2bfa">

    <set-variable value="{search.status}" doc:name="Search Status" doc:id="4280b5af-
7754-418f-ba29-4a6bf6e4d20a" variableName="searchStatus" />
    <mule4-medijs-connector:search-purchase-orders
        doc:name="Search Purchase Orders"
        doc:id="d73a64ec-d950-406a-bd46-cc6776acd274"
        config-ref="Medius_Connector_Mule4_Config"
        statusUriParam="#[upper(vars.searchStatus)]"/>
    <choice doc:name="Choice"
        doc:id="3590f73d-bad0-4946-9d30-2f3faa02924c">
        <when expression="#[sizeOf(payload) > 0]">
            <set-variable value="{message.element.position}"
                doc:name="Element Position"
                doc:id="98a3cd7a-7894-4cfc-b1e8-7f66a0897775"
                variableName="pos" />
            <choice doc:name="Choice"
                doc:id="ec801736-747a-48d6-98be-0f9bf092364b">
                <when
                    expression="#[vars.pos >= 0 and
sizeOf(payload) > vars.pos]">
                    <set-variable
                        value="#[output application/json ---
payload[vars.pos as Number]]"
                        doc:name="Get and Set Element"
                        doc:id="78c90bf0-7ea2-49d3-ae6b-
ef89db8322ca"
                        variableName="elementValue" />
                    <set-variable
                        value="#[vars.elementValue.url
splitBy ("/")]'"
                        doc:name="Message Status Link
Reference Value Split By '/'"
                        doc:id="c61e65b6-4669-4662-a281-
1dc8020d3332" variableName="link" />
                    <set-variable value="#[vars.link[-1]]"
                        doc:name="Message Id"
                        doc:id="f745cde6-5ffa-48ef-8152-
b4ba0120820f"
                        variableName="messageId" />
                    <set-variable value="#[vars.link[-2]]"
                        doc:name="Message Status"

```

|                                               |                                              |
|-----------------------------------------------|----------------------------------------------|
| a347c6bfd5d6"                                 | doc:id="e00d827d-da01-4491-8ebb-             |
|                                               | variableName="messageStatus" />              |
| order-request-message-status                  | <mule4-medijs-connector:get-purchase-        |
| Request Message Status"                       | doc:name="Get Purchase Order                 |
| d3e2a7113a26"                                 | doc:id="712a0fe3-a0ef-4943-99a3-             |
| ref="Medius_Connector_Mule4_Config"           | config-                                      |
| messageIdUriParam="#[vars.messageId]" />      | <logger level="INFO" doc:name="Logger"       |
| f7b5316b1bdb"                                 | doc:id="26daa9a7-3c2a-40c1-8405-             |
|                                               | message="#[payload]" />                      |
| order-request-message-content                 | <mule4-medijs-connector:get-purchase-        |
| Request Message Content"                      | doc:name="Get Purchase Order                 |
| 6d0872628a8f"                                 | doc:id="060d04b5-1080-4914-9322-             |
| ref="Medius_Connector_Mule4_Config"           | config-                                      |
| messageStatusUriParam="#[vars.messageStatus]" |                                              |
| messageIdUriParam="#[vars.messageId]" />      | <set-variable                                |
| value="#[payload.CorrelationKey]"             | doc:name="Correlation Key"                   |
| c5aaa55d8963"                                 | doc:id="317e457c-eb6b-4b4a-8ce9-             |
|                                               | variableName="corKey" />                     |
| 5888b7dc151a"                                 | <logger level="INFO" doc:name="Logger"       |
|                                               | doc:id="2a68855d-d1a9-4b66-ba83-             |
|                                               | message="#[payload]" />                      |
| a07204e24925"                                 | <set-variable value="\$ {new.update.status}" |
|                                               | doc:name="New Update Status"                 |
|                                               | doc:id="1ac0d3e1-8eb2-4139-be7a-             |
| order-message-status                          | variableName="newStatus" />                  |
| Message Status"                               | <mule4-medijs-connector:update-purchase-     |
| cc0412e974ec"                                 | doc:name="Update Purchase Order              |
|                                               | doc:id="9b8debac-86f4-4cea-853c-             |

|                                                                          |                                           |
|--------------------------------------------------------------------------|-------------------------------------------|
| ref="Medius_Connector_Mule4_Config"                                      | config-                                   |
| messageIdUriParam="#[vars.messageId]"                                    |                                           |
| messageStatusUriParam="#[vars.messageStatus]">                           |                                           |
| purchase-order-message-status-body><![CDATA[#[%dw 2.0                    | <mule4-medijs-connector:update-           |
| output application/json                                                  |                                           |
| ---                                                                      |                                           |
| {                                                                        |                                           |
| "NewStatus": vars.newStatus                                              |                                           |
| ]]]]></mule4-medijs-connector:update-purchase-order-message-status-body> |                                           |
| order-message-status>                                                    | </mule4-medijs-connector:update-purchase- |
| 9a77c66de8a4"                                                            | <logger level="INFO" doc:name="Logger"    |
|                                                                          | doc:id="2c0cd63f-2a10-4b29-99b1-          |
|                                                                          | message="#[payload]" />                   |
| purchase-order-statuses                                                  | <mule4-medijs-connector:bulk-update-      |
| Order Statuses"                                                          | doc:name="Bulk Update Purchase            |
| 63a1f4637b6c"                                                            | doc:id="66369a8b-fe88-4b72-8f59-          |
| ref="Medius_Connector_Mule4_Config"                                      | config-                                   |
| statusUriParam="#[upper(vars.newStatus)]"                                |                                           |
| correlationKeyFilterQueryParam="#[vars.corKey]">                         |                                           |
| update-purchase-order-statuses-body><![CDATA[#[%dw 2.0                   | <mule4-medijs-connector:bulk-             |
| output application/json                                                  |                                           |
| ---                                                                      |                                           |
| {                                                                        |                                           |
| "NewStatus": vars.messageStatus                                          |                                           |
| ]]]]></mule4-medijs-connector:bulk-update-purchase-order-statuses-body>  |                                           |
| purchase-order-statuses>                                                 | </mule4-medijs-connector:bulk-update-     |
| 415475e6f917"                                                            | <logger level="INFO" doc:name="Logger"    |
|                                                                          | doc:id="c24ce648-6e0f-4150-b360-          |
|                                                                          | message="#[payload]" />                   |
|                                                                          | </when>                                   |
|                                                                          | </choice>                                 |
|                                                                          | </when>                                   |
|                                                                          | </choice>                                 |
| </sub-flow>                                                              |                                           |

```

<sub-flow name="purchase-order-request-confirmation-flow"
  doc:id="f2363ce9-ebae-4aed-95ab-a7b287897929">

  <set-variable value='${correlation.key}'
    doc:name="Correlation Key"
    doc:id="71ad74ef-96fe-46f7-9bcf-b876de27b857"
    variableName="correlationkey" />

  <mule4-medijs-connector:purchase-order-request-confirmation doc:name="Purchase
Order      Request      Confirmation"      doc:id="8cdb158f-0157-4803-9d93-007b332519ff"      config-
ref="Medius_Connector_Mule4_Config">
    <mule4-medijs-connector:purchase-order-request-confirmation-
body><![CDATA[#[%dw 2.0
  output application/json
  ---
  {
    "CorrelationKey": vars.correlationkey,
    "ResponseData": {
    }
  }
]]]></mule4-medijs-connector:purchase-order-request-confirmation-body>
    </mule4-medijs-connector:purchase-order-request-confirmation>
    <logger level="INFO" doc:name="Response Headers" doc:id="7acdee2d-b23d-462a-
a920-db6a331b66d7" message="#[attributes]" />
    <choice doc:name="Choice"
      doc:id="4d9265b2-03c7-4b0e-990e-506ea35718ba">
      <when
        expression="#[attributes.statusCode == 201]">
        <logger level="INFO" doc:name="Logger" doc:id="540a3403-3c6d-
4b99-89bf-b9e984cbc3d2" message="Purchase Order Request Confirmation is done successfully." />
        <set-variable value="#[attributes.headers.location splitBy ('/'])"
doc:name="Location Header Value Split By /" doc:id="73a4748d-0011-4c3a-914c-6d05ad9b076d"
variableName="location"/>
        <set-variable value="#[vars.location[-1]]" doc:name="Message Id"
doc:id="e58e7a06-c45d-4f6b-bcaf-1750d7457d74" variableName="messageId"/>
        <mule4-medijs-connector:get-purchase-order-request-message-
status doc:name="Get Purchase Order Request Message Status" doc:id="ce215332-e4d5-4dc7-a3b4-
ac816fec290d" config-ref="Medius_Connector_Mule4_Config" messageIdUriParam="#[vars.messageId]" />
        <logger level="INFO" doc:name="Response Headers"
doc:id="4566c155-71b4-4aff-a4a0-d107471094e1" message="#[attributes]" />
        <logger level="INFO" doc:name="Payload" doc:id="cbd1b92d-9a9e-
4635-baab-9b1290687cfe" message="#[payload]" />
        <choice doc:name="Choice"
          doc:id="65ac9eb1-594a-4d89-9272-f525b4ad6ed1">
          <when
            expression="#[attributes.statusCode == 200]">
            <set-variable value="#[payload.status]"
doc:name="Message Status" doc:id="8d60f511-a860-40ea-afdf-232b22919053"
variableName="msgStatus"/>
            <mule4-medijs-connector:get-purchase-
order-request-message-content doc:name="Get Purchase Order Request Message Content"

```

```

doc:id="e9fcc1b8-faae-4289-af1f-782125c28e87"          config-ref="Medius_Connector_Mule4_Config"
messageStatusUriParam="#[vars.msgStatus]" messageIdUriParam="#[vars.messageId]"/>
    </when>
    </choice>
</when>
</choice>
</sub-flow>

<sub-flow
    name="purchase-order-request-cancel-confirmation-flow"
    doc:id="b148e980-708c-4dac-82c8-3e4375232adb">

    <set-variable value='${correlation.key}'
        doc:name="Correlation Key"
        doc:id="67ab2f1d-6f8d-47c9-8de7-a9e8a74770b4"
        variableName="correlationkey" />

    <mule4-medijs-connector:purchase-order-cancellation-confirmation
doc:name="Purchase Order Cancellation Confirmation" doc:id="6b615aee-9261-4652-8cd3-57c20dca8467"
config-ref="Medius_Connector_Mule4_Config">
        <mule4-medijs-connector:purchase-order-cancellation-
confirmation-body><![CDATA[#[%dw 2.0
output application/json
---
{
    "CorrelationKey": vars.correlationkey,
    "ResponseData": {
    }
}]]]></mule4-medijs-connector:purchase-order-cancellation-confirmation-body>
        </mule4-medijs-connector:purchase-order-cancellation-confirmation>
        <logger level="INFO" doc:name="Response Headers" doc:id="d870d0ca-29d5-4504-
ab70-8095b7b31895" message="#[attributes]" />
    </sub-flow>

</mule>

```

## External System Procurement Purchase Order Confirmation and Search, Fetch, Update Status

This use case demonstrates the usage of various External System Purchase Order operations for example Search, Fetch Message Status, Fetch Message Content, Update Specific as well as Bulk Update Messages status. This demo will show confirmation of order and fetch message content from response headers to verify data flow based on selection of flow to execute.

- In Anypoint Studio, click **File > New > Mule Project**, name the project **mule-medijs-connector-**

**external-sys-purchase-order-demo** and click Finish.

- In the search field, type **Http** and drag the **HTTP Listener** to the canvas.
- Click the **HTTP Listener**, click the green plus sign to the right of Connector Configuration. On the next screen, add the host and port, click OK
- Provide **Path** to HTTP Listener (Example: **/external-sys-purchase-order**)
- On the Mule Palette, search for **Set Variable** and drag it onto the canvas. Set value of variable from **mule-application.properties** file key “**selected.option**” as **\${selected.option}** and give name of variable as **selectedOpt** and display name as **Selected Option**.
- There are two options available for execution of two different flows. This option user can change from **mule-application.properties** file.
  - **MSG\_SEARCH\_FETCH\_UPDATE** - execute Search, Fetch Status, Fetch Content, Update Status and Bulk Update Status operations related flow
  - **CONFIRMATION** - Send External System Procurement Purchase Order record to ERP and then Confirmation flow.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- There will be two **When** blocks for added **Choice**. Those will be as follows.
- Add below condition as expression in **Choice** to check if user wants to execute Search, Fetch Status, Fetch Content, Update Status and Bulk Update Status operations related flow.

```
vars.selectedOpt == 'MSG_SEARCH_FETCH_UPDATE'
```

- Add **Flow Reference** to call another sub flow '**external-sys-purchase-order-message-flow**' to Search, Fetch Status, Fetch Content, Update Status and Bulk Update Status operations related flow.
- Add below condition as expression in **Choice** to check if the user wants to execute External System Procurement Goods Receipt Note Confirmation operation related flow.

```
vars.selectedOpt == 'CONFIRMATION'
```

- Add **Flow Reference** to call another sub flow '**external-sys-purchase-order-confirmation-flow**' to execute Send Record to ERP and then External System Procurement Purchase Order Confirmation operation related flow
- Create a new sub flow **external-sys-purchase-order-message-flow**.
- Add **Set Variable** and set variable name as **searchStatus** and value as **\${search.status}** key from **mule-application.properties** file and display name as **Search Status**.
- Add **Search External System Purchase Orders** Medius operation from **Mule Palette** to canvas and provide **status** as expression **upper(vars.searchStatus)**. This operation returns all messages

that match the parameters sent in. The mandatory parameter is the status parameter, besides that the other parameters are sent in as query string parameters to narrow down the result.

- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if payload has record.

```
sizeOf(payload) > 0
```

- Add **Set Variable** and set variable name as **pos** and value as **\${message.element.position}** key from **mule-application.properties** file and display name as **Element Position**. It will say which record user wants to process further based on position. For the first record its value will be 0, for the second record it will have 1 and so on.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if selected position is greater than or equal to zero and size of payload has record for that particular position.

```
vars.pos >= 0 and sizeOf(payload) > vars.pos
```

- Add **Set Variable** and set variable name as **elementValue** and value as expression **output application/json --- payload[vars.pos as Number]** and display name as **Get and Set Element**. It will fetch the element from a particular position.
- Add **Set Variable** and set variable name as **link** and value as expression **vars.elementValue.url splitBy ("/")** and display name as **Message Status Link Reference Value Split By /**. It will be useful to set a message Id.
- Add **Set Variable** and set variable name as **messageId** and value as expression **vars.link[-1]** and display name as **Message Id** here -1 means last element from array and it is messageId.
- Add **Set Variable** and set variable name as **messageStatus** and value as expression **vars.link[-2]** and display name as **Message Status** here -2 means the second last element from array and it is messageStatus.
- Search **Get External System Purchase Order Message Status** Medius operation from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId**. The operation with message id as input returns a status and an effective link to purchase order content. This method allows you to check the status of a given message.
- Add **Logger** to log response and set message as expression **payload**.
- Search **Get External System Purchase Order Message Content** from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId** and Message Status as expression **vars.messageStatus**. This operation returns the purchase order content from a message that matches the

provided status and id. If the message id is in another status, nothing is returned.

- Add **Set Variable** and set variable name as **corKey** and value as expression **payload.CorrelationKey** and display name as **Correlation Key**
- Add **Logger** to log response and set the message as expression **payload**.
- Add **Set Variable** and set variable name as **newStatus** and value as **\${new.update.status}** from mule properties file and display name as **New Update Status**. This indicates a new update status for a given record. It will be useful in further operations.
- Search **Update External System Purchase Order Message Status** from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId** and Message Status as expression **vars.messageStatus** and provide **body** parameter as expression as follows:

```
%dw 2.0
output application/json
---
{
    "NewStatus": vars.newStatus
}
```

- This operation updates the status of a purchase order message when processing it.
- Add **Logger** to log response and set message as expression **payload**
- Search **Bulk Update External System Purchase Order Statuses** from **Mule Palette** and drag to canvas and provide Status as expression **upper(vars.newStatus)** and Correlation Key Filter as expression **vars.corKey** and provide **body** parameter as expression as follows:

```
%dw 2.0
output application/json
---
{
    "NewStatus": vars.messageStatus
}
```

- This operation updates the status for all messages that meet the filter parameters.
- Add **Logger** to log response and set the message as expression **payload**.
- Create a new sub flow **external-sys-purchase-order-confirmation-flow**.
- Add **Set Variable** and set variable name as **correlationKey** and value as **\${correlation.key}** key from **mule-application.properties** file and display name as **Correlation Key**.
- Add **Set Variable** and set variable name as **searchStatus** and value as **\${search.status}** key from **mule-**

**application.properties** file. and display name as **Search Status**

- Add **Search External System Purchase Orders** Medius operation from **Mule Palette** to canvas and provide **status** as expression **upper(vars.searchStatus)** and **Correlation Key Filter** as expression **vars.correlationKey**. This operation returns all messages that matches the parameters sent in. The mandatory parameter is the status parameter, besides that the other parameters are sent in as query string parameters to narrow down the result.
- Add **Logger** to log searched records and set message as expression **payload**
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if payload has any records.

`sizeof(payload) > 0`

- Add **Set Variable** and set variable name as **pos** and value as **\${message.element.position}** key from **mule-application.properties** file and display name as **Element Position**. It will say which record user wants to process further based on position. For the first record its value will be 0, for the second record it will have 1 and so on.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if selected position is greater than or equal to zero and size of payload has record for that particular position.

`vars.pos >= 0 and sizeof(payload) > vars.pos`

- Add **Set Variable** and set variable name as **elementValue** and value as expression **output application/json --- payload[vars.pos as Number]** and display name as **Get and Set Element**. It will fetch the element from a particular position.
- Add **Set Variable** and set variable name as **link** and value as expression **vars.elementValue.url splitBy ("/")** and display name as **Message Status Link Reference Value Split By /**. It will be useful to set a message Id and message status.
- Add **Set Variable** and set variable name as **messageId** and value as expression **vars.link[-1]** and display name as **Message Id**. Here -1 means the last element from the array and it is messageId.
- Add **Set Variable** and set variable name as **messageStatus** and value as expression **vars.link[-2]** and display name as **Message Status**. Here -2 means the second last element from the array and it is messageStatus.
- Search **Get External System Purchase Order Message Content** from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId** and Message Status as expression

**vars.messageStatus** This operation returns the purchase order content from a message that matches the provided status and id. If the message id is in another status, nothing is returned.

- Add **Logger** to log response and set message as expression **payload**
- Add **Set Variable** and set variable name as **messageContent** and value as expression **payload** and display name as **Message Content**.
- Add **Flow Reference** to call another sub flow '**send-purchase-order-to-ERP-and-confirmation-flow**' to process the fetched record to send it to ERP and then do confirmation on Medius by calling operation.
- Create a new **Sub Flow** named **send-purchase-order-to-ERP-and-confirmation-flow**.
- Add **try** block by dragging from **Mule Palette** and give name **Try - Block for Sending Purchase Order to ERP** and keep settings default one . This block will be used for sending records to the ERP system. Right now it is not having anything except setting variables to decide which flow we need to execute further. This block can be revised as per requirement.
- Add **Set Variable** and set variable name as **isSendToERPSuccessfull** and value as expression **output application/json --- true as Boolean** and display name as **Success Flag**. Right now it is hard-coded but it can be set based on response received from send to ERP flow.
- In the Error Handling block, add **On Error Propagate** from **Mule Palette** and provide type as **ANY**
- In **On Error Propagate** block, add **Logger** to log error and set display name as **Error** and message as expression **error**
- Add **Transform Message** from **Mule Palette** and add the following in that.

```
%dw 2.0
output application/java
---
write(error, "application/json")
```

- Add **Set Variable** and set display name as **Error**, variable name as **errorRes** and value as expression **payload**.
- Add **Logger** to log error response variable and set display name as **Error Message** and message as expression **vars.errorRes**
- Add **Post A Procurement Error** Medius operation from **Mule Palette** and provide **body** as follows:

```
%dw 2.0
output application/json
---
{
  "Errors": [
    {
```

```

    "Message": vars.errorRes
  }
],
  "CorrelationKey": vars.correlationKey
}

```

- This method accepts an error message object, including a list of error messages. The error message object includes a CorrelationKey that needs to match the corresponding purchase order message.
- Add **Logger** to log response attributes and set display name as **Response Headers** message as expression **attributes**.
- After the **Try** block, add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if sending to ERP is successful or not.

```
vars.isSendToERPSuccessfull == true
```

- Add **Logger** to log success message and set message **Send Purchase Order to ERP is successfully done..**
- Add **Transform Message** from **Mule Palette** and add the following in that.

```

%dw 2.0
output application/json
---
{
  OrderNumber: vars.messageContent.orderNumber,
  Lines: vars.messageContent.lines map {
    LineNumber: $.lineNumber
  }
}

```

- Add **Logger** to log response attributes and set the message as expression **payload**.
- Add **Purchase Order To Erp Or Goods Receipt Note Confirmation** Medius operation from **Mule Palette** and set body parameter as follows:

```

%dw 2.0
output application/json
---
{

```

```

CorrelationKey: vars.correlationKey,
PurchaseOrderConfirmation: {
    OrderNumber: payload.OrderNumber,
    Lines: payload.Lines
}
}

```

- This operation accepts a purchase-order-to-ERP confirmation with or without the goods-receipt-note part as a response to either the purchase-order-to-ERP or the goods-receipt-note confirmation or cancellation message, that was retrieved in the earlier step. The confirmation object includes a **CorrelationKey** that needs to match the corresponding purchase-order-to-ERP or goods-receipt-note message.
- Add **Logger** to log response attributes and set message as expression **attributes**
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check the status code of previous confirmation operation from **Medius**.

```
attributes.statusCode == 201
```

- Add **Logger** to log success message and set display name as **Confirmation Success** and message as **Purchase order to ERP or Goods Receipt Note Confirmation is done successfully**.
- Add **Set Variable** and set display name as **Location Header Value Split By /**, variable name as **location** and value as expression **attributes.headers.location splitBy ("/")**.
- Add **Set Variable** and set variable name as **messageId** and value as expression **vars.location[-1]** and display name as **Message Id**. Here -1 means the last element from the array and it is messageId.
- Search **Get External System Purchase Order Message Status** Medius operation from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId**. The operation with message id as input returns a status and an effective link to purchase order content. This method allows you to check the status of a given message.
- Add **Logger** to log response attributes and set the message as expression **attributes** with display name as **Response Headers**.
- Add **Logger** to log response and set message as expression **payload** and display name as **Response**.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check the status code of previous confirmation operation from **Medius**.

```
attributes.statusCode == 200
```

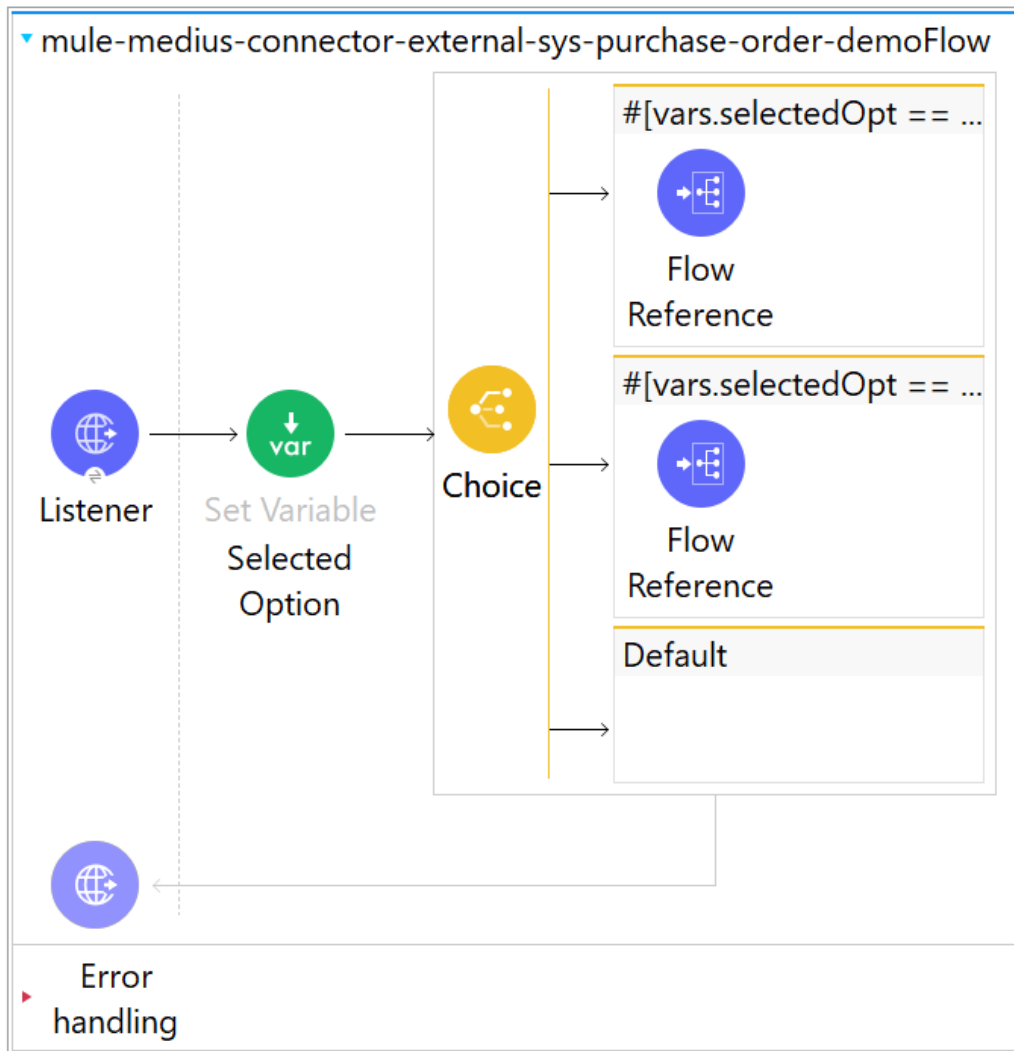
- Add **Set Variable** and set display name as **Message Status**, variable name as **msgStatus** and value as expression **payload.status**.
- Search **Get External System Purchase Order Message Content** from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId** and Message Status as expression **vars.msgStatus**. This operation returns the purchase order content from a message that matches the provided status and id. If the message id is in another status, nothing is returned.
- Add **Try** block with display name as **Try - Block to Raise Error and Log in Medius** if **vars.isSendToERPSuccessfull == false** means in default block of that **Choice** block.
- Add **Raise Error** from **Mule Palette** in **Try** block and provide type as **ANY** and description as **Send Purchase Order to ERP is failed..**
- In the Error Handling block, add **On Error Propagate** from **Mule Palette** and provide type as **ANY**
- In **On Error Propagate** block, add **Logger** to log error and set display name as **Error** and message as expression **error**
- Add **Set Variable** and set display name as **Error**, variable name as **errorRes** and value as expression **error.detailedDescription**.
- Add **Logger** to log error response variable and set display name as **Error Message** and message as expression **vars.errorRes**
- Add **Post A Procurement Error** Medius operation from **Mule Palette** and provide **body** as follows:

```
%dw 2.0
output application/json
---
{
  "Errors": [
    {
      "Message": vars.errorRes
    }
  ],
  "CorrelationKey": vars.correlationKey
}
```

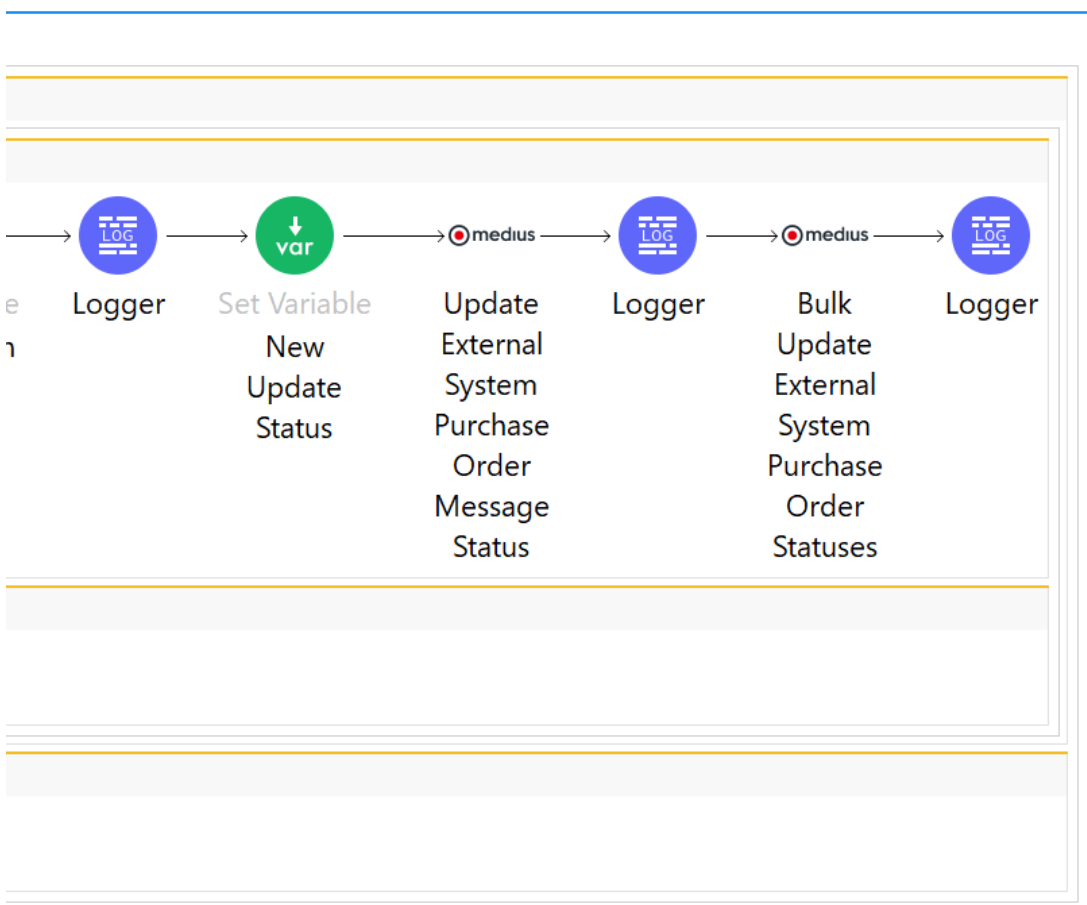
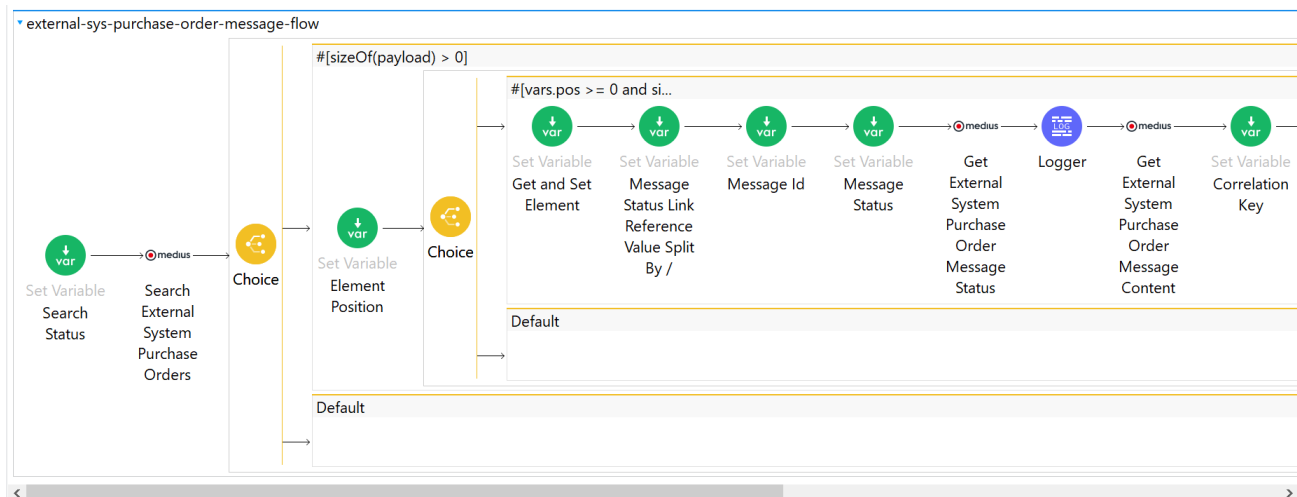
- This method accepts an error message object, including a list of error messages. The error message object includes a **CorrelationKey** that needs to match the corresponding purchase order message.
- Add **Logger** to log response attributes and set display name as **Response Headers** message as expression **attributes**

- After creating the flow, right-click the project and click **Run As > Mule Application**.
- Once the application is deployed, use **HTTP Listener's Path** to execute the Mule Flow.

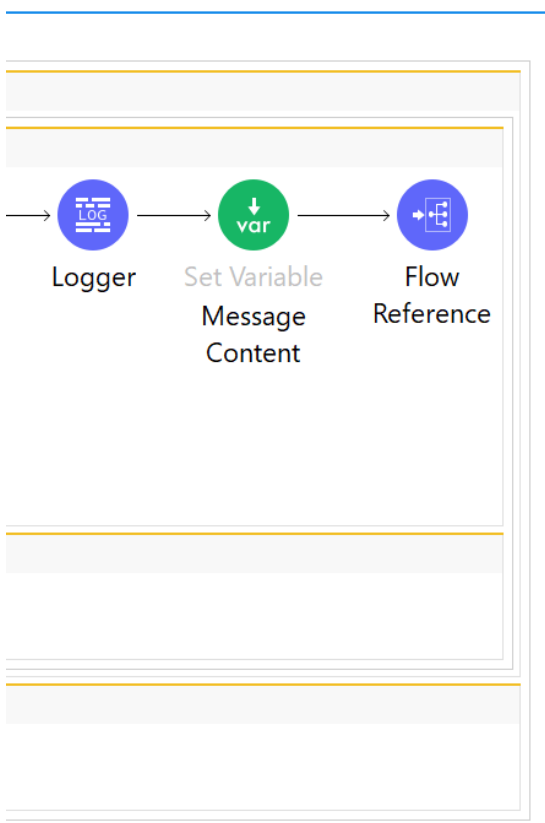
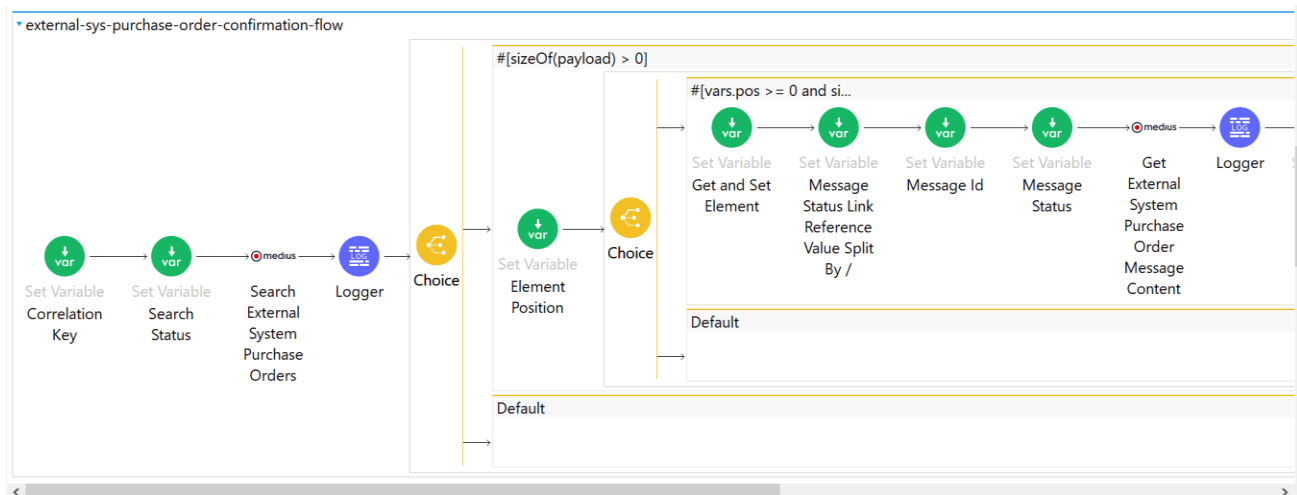
#### mule-mediuis-connector-external-sys-purchase-order-demoFlow



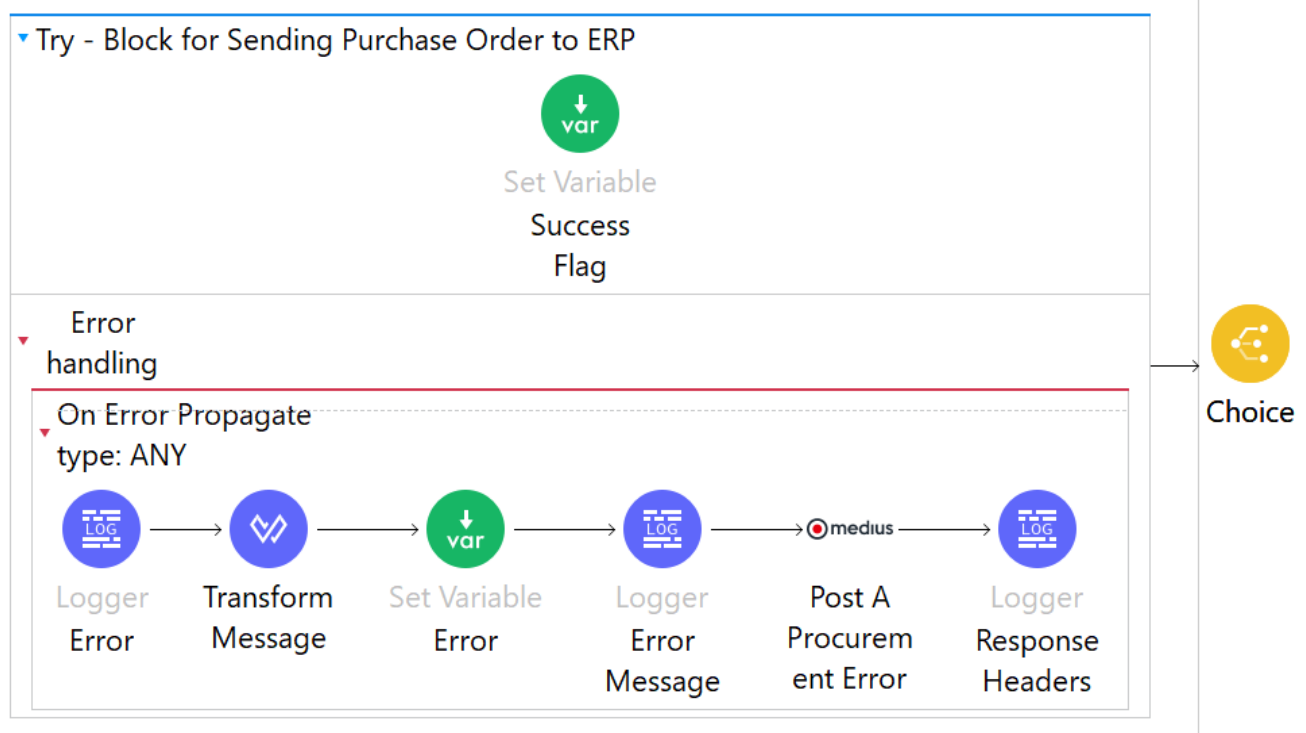
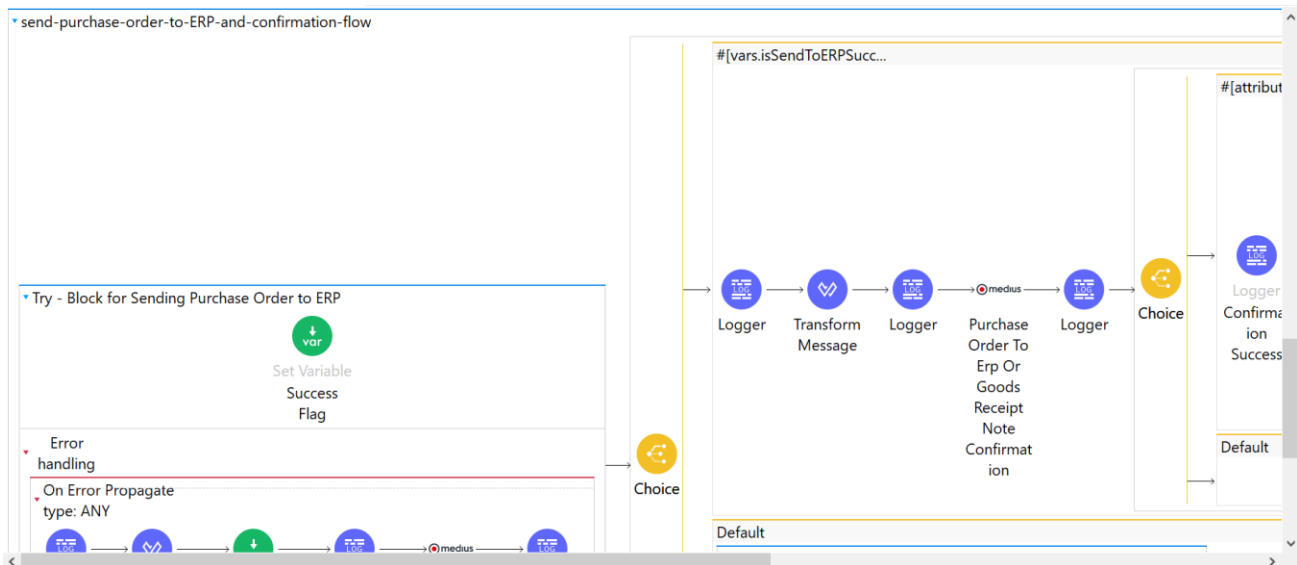
#### external-sys-purchase-order-message-flow

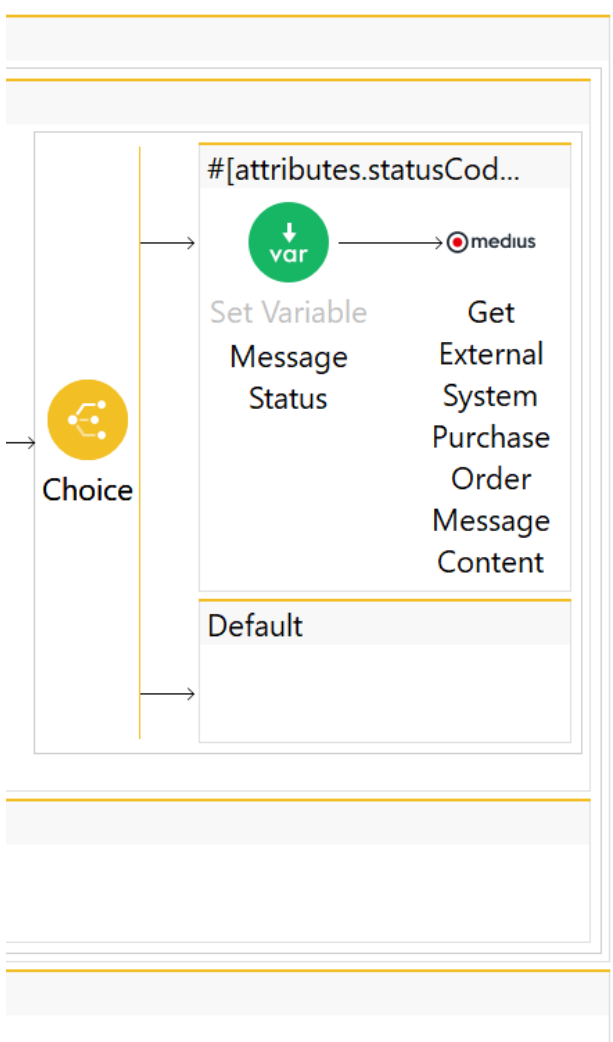
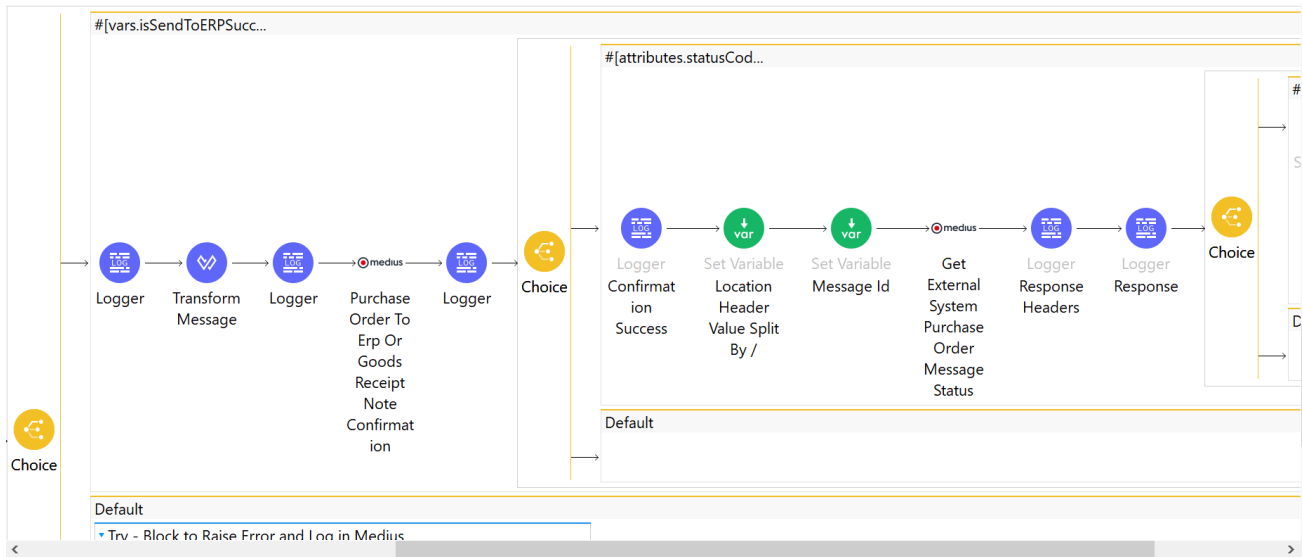


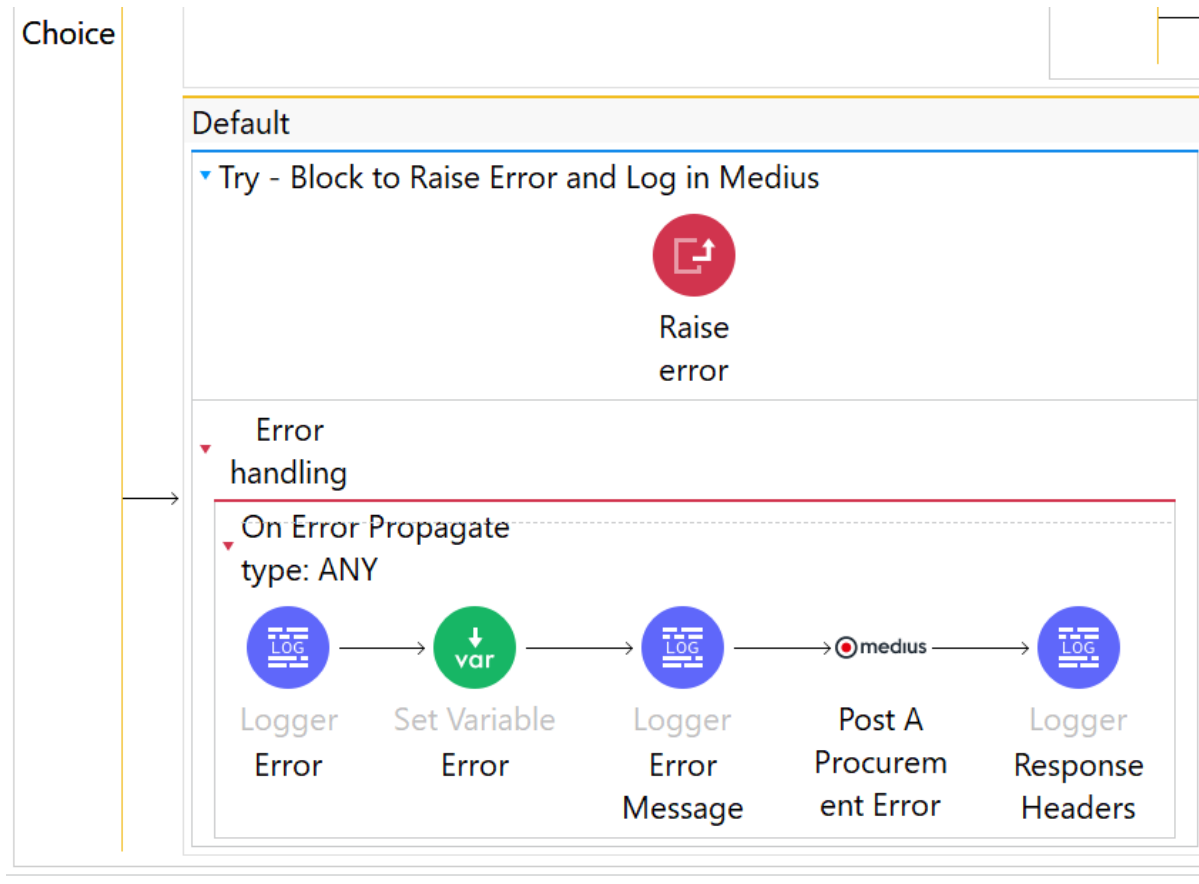
external-sys-purchase-order-confirmation-flow



**send-purchase-order-to-ERP-and-confirmation-flow**







**Use Case Configuration XML :** Paste this XML code into Anypoint Studio to experiment with the flow described above.

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

<mule xmlns:ee="http://www.mulesoft.org/schema/mule/ee/core"
      xmlns:mule4-mediuss-connector="http://www.mulesoft.org/schema/mule/mule4-mediuss-connector"
      xmlns:http="http://www.mulesoft.org/schema/mule/http"
      xmlns="http://www.mulesoft.org/schema/mule/core"
      xmlns:doc="http://www.mulesoft.org/schema/mule/documentation"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:schemaLocation="http://www.mulesoft.org/schema/mule/core
http://www.mulesoft.org/schema/mule/core/current/mule.xsd
      http://www.mulesoft.org/schema/mule/http
http://www.mulesoft.org/schema/mule/http/current/mule-http.xsd
      http://www.mulesoft.org/schema/mule/mule4-mediuss-connector
http://www.mulesoft.org/schema/mule/mule4-mediuss-connector/current/mule-mule4-mediuss-connector.xsd
      http://www.mulesoft.org/schema/mule/ee/core
http://www.mulesoft.org/schema/mule/ee/core/current/mule-ee.xsd">
```

```

<configuration-properties
    file="mule-application.properties" />

<http:listener-config name="HTTP_Listener_config"
    doc:name="HTTP Listener config"
    doc:id="09f5cca2-7cd3-437e-a268-5ee0c7824a3d">
    <http:listener-connection host="0.0.0.0"
        port="8081" />
</http:listener-config>

<mule4-medijs-connector:config
    name="Medius_Connector_Mule4_Config"
    doc:name="Medius Connector Mule4 Config"
    doc:id="1adba4ea-02c5-4d97-a897-a0b14afe87be">
    <mule4-medijs-connector:medijsflowclouauth-connection
        baseUri="{medijs.base.url}">
        <mule4-medijs-connector:oauth-client-credentials
            clientId="{medijs.client.id}" tokenUrl="{medijs.token.url}"
            clientSecret="{medijs.client.secret}" scopes="{medijs.scopes}"
        />
    />
</mule4-medijs-connector:medijsflowclouauth-connection>
</mule4-medijs-connector:config>

<flow name="mule-medijs-connector-external-sys-purchase-order-demoFlow"
    doc:id="44b03b34-9f23-49a9-b06c-359d8bc8bac2">
    <http:listener doc:name="Listener"
        doc:id="a25dfd77-e60b-4e52-9d74-97b820122e98"
        config-ref="HTTP_Listener_config" path="/external-sys-purchase-order">
    </http:listener>

    <set-variable value="{selected.option}"
        doc:name="Selected Option"
        doc:id="4525e3bb-2cf1-4812-a1f8-5866676e17ed"
        variableName="selectedOpt" />

    <choice doc:name="Choice"
        doc:id="29e83357-896a-4227-9944-3870ba99a548">
        <when
            expression="#[vars.selectedOpt ==
'MSG_SEARCH_FETCH_UPDATE']">

            <flow-ref doc:name="Flow Reference"
                doc:id="60b14796-6b2a-4d16-9227-0d4247713d63"
                name="external-sys-purchase-order-message-flow" />
        </when>
        <when expression="#[vars.selectedOpt == 'CONFIRMATION']">

            <flow-ref doc:name="Flow Reference"
                doc:id="6252c0ca-de42-4d3f-8bbf-2d69137f8edd"

```

```

name="external-sys-purchase-order-confirmation-flow"
/>
</when>
</choice>
</flow>

<sub-flow name="external-sys-purchase-order-message-flow"
doc:id="c2513693-0ed2-4527-a88e-4b06a9e82c6a">
<set-variable value="\${search.status}"
doc:name="Search Status"
doc:id="2063372a-23ab-464d-92e8-a5138d9f705e"
variableName="searchStatus" />
<mule4-medijs-connector:search-external-system-purchase-orders
doc:name="Search External System Purchase Orders"
doc:id="36d32057-5e05-49b3-a343-c644adc360bd"
config-ref="Medius_Connector_Mule4_Config"
statusUriParam="\#[upper(vars.searchStatus)]" />
<choice doc:name="Choice"
doc:id="1ba099a3-f30b-4745-89e5-b5770cd0f565">
<when expression="\#[sizeOf(payload) > 0]">
<set-variable value="\${message.element.position}"
doc:name="Element Position"
doc:id="26aa86ea-d211-42f3-8ddf-4b25661729fb"
variableName="pos" />
<choice doc:name="Choice"
doc:id="460d8681-ee4b-436a-ab28-af847c6fbfd3">
<when
expression="\#[vars.pos >= 0 and
sizeOf(payload) > vars.pos]">
<set-variable
value="\#[output application/json ---
payload[vars.pos as Number]]"
doc:name="Get and Set Element"
doc:id="3f2b7fc0-b75a-46c8-b116-
0b5160f7288e"
variableName="elementValue" />
<set-variable
value='\#[vars.elementValue.url
splitBy ("/")]'
doc:name="Message Status Link"
doc:id="98df4332-34db-458d-86a9-
4ab806da04ed" variableName="link" />
<set-variable value="\#[vars.link[-1]]"
doc:name="Message Id"
doc:id="a23e0f80-10de-4f70-9922-
a6e2ebbc3cd1"
variableName="messageId" />
<set-variable value="\#[vars.link[-2]]"
doc:name="Message Status"

```

|                                               |                                              |
|-----------------------------------------------|----------------------------------------------|
| 63cf5258952a"                                 | doc:id="5b4607f9-cac5-4f63-ba9e-             |
|                                               | variableName="messageStatus" />              |
| system-purchase-order-message-status          | <mule4-medi-us-connector:get-external-       |
| Purchase Order Message Status"                | doc:name="Get External System                |
| 7b290c88bc8d"                                 | doc:id="f299ad1d-a380-47e0-a14f-             |
| ref="Medius_Connector_Mule4_Config"           | config-                                      |
| messageIdUriParam="#[vars.messageId]" />      |                                              |
| 8a4cfd22ef4b"                                 | <logger level="INFO" doc:name="Logger"       |
|                                               | doc:id="dbb17b81-8e7f-41fe-a854-             |
|                                               | message="#[payload]" />                      |
| system-purchase-order-message-content         | <mule4-medi-us-connector:get-external-       |
| Purchase Order Message Content"               | doc:name="Get External System                |
| 00670780a97a"                                 | doc:id="58547295-b7be-44eb-b4f8-             |
| ref="Medius_Connector_Mule4_Config"           | config-                                      |
| messageStatusUriParam="#[vars.messageStatus]" |                                              |
| messageIdUriParam="#[vars.messageId]" />      |                                              |
| value="#[payload.CorrelationKey]"             | <set-variable                                |
| 1950d6f56e5f"                                 | doc:name="Correlation Key"                   |
|                                               | doc:id="0ef7dec5-afb9-4c3f-8756-             |
|                                               | variableName="corKey" />                     |
|                                               | <logger level="INFO" doc:name="Logger"       |
|                                               | doc:id="84a89c12-247d-4d19-a4f4-             |
| 79a9460ba314"                                 | message="#[payload]" />                      |
|                                               | <set-variable value="\$ {new.update.status}" |
|                                               | doc:name="New Update Status"                 |
|                                               | doc:id="86f6bdeb-5f0d-4a3a-a898-             |
| 8970cd573981"                                 | variableName="newStatus" />                  |
| system-purchase-order-message-status          | <mule4-medi-us-connector:update-external-    |
| Purchase Order Message Status"                | doc:name="Update External System             |
| ab55e22bd09a"                                 | doc:id="50d38232-4dc8-4852-aa55-             |

```

config-
ref="Medius_Connector_Mule4_Config"

    messageIdUriParam="#[vars.messageId]"

    messageStatusUriParam="#[vars.messageStatus]">
<mule4-medijs-connector:update-
external-system-purchase-order-message-status-body><![CDATA[#[%dw 2.0
output application/json
---
{
    "NewStatus": vars.newStatus
}]]]></mule4-medijs-connector:update-external-system-purchase-order-message-status-body>
</mule4-medijs-connector:update-external-
system-purchase-order-message-status>
<logger level="INFO" doc:name="Logger"
doc:id="d686f4bb-986e-4661-ad8f-
80ed3125d18c"
message="#[payload]" />
<mule4-medijs-connector:bulk-update-
external-system-purchase-order-statuses
doc:name="Bulk Update External
System Purchase Order Statuses"
doc:id="540b1034-de11-4486-9061-
8512b1a0123c"
config-
ref="Medius_Connector_Mule4_Config"

    statusUriParam="#[upper(vars.newStatus)]"

    correlationKeyFilterQueryParam="#[vars.corKey]">
<mule4-medijs-connector:bulk-
update-external-system-purchase-order-statuses-body><![CDATA[#[%dw 2.0
output application/json
---
{
    "NewStatus": vars.messageStatus
}]]]></mule4-medijs-connector:bulk-update-external-system-purchase-order-statuses-
body>
</mule4-medijs-connector:bulk-update-
external-system-purchase-order-statuses>
<logger level="INFO" doc:name="Logger"
doc:id="545d0c54-3aa2-4b4b-a966-
758619bd1570"
message="#[payload]" />
</when>
</choice>
</when>
</choice>
</sub-flow>

```

```

<sub-flow name="external-sys-purchase-order-confirmation-flow"
  doc:id="adf749ed-4f68-4c74-9b78-7a9ef91c03fd">

  <set-variable value="{correlation.key}"
    doc:name="Correlation Key"
    doc:id="627bdb52-c838-4f40-976d-e2a4522a23ed"
    variableName="correlationKey" />
  <set-variable value="{search.status}"
    doc:name="Search Status"
    doc:id="27d3d240-d20f-4bba-84e5-ee7c6768c7a8"
    variableName="searchStatus" />
  <mule4-medijs-connector:search-external-system-purchase-orders
    statusUriParam="#[upper(vars.searchStatus)]"
    doc:name="Search External System Purchase Orders"
    doc:id="73457320-07e4-410f-a739-4541f49a55bf"
    config-ref="Medius_Connector_Mule4_Config"
    correlationKeyFilterQueryParam="#[vars.correlationKey]" />
  <logger level="INFO" doc:name="Logger"
    doc:id="d1ff6626-b0ad-4b30-8087-5db9b9404f1f" message="#[payload]" />
  <choice doc:name="Choice"
    doc:id="3f6ccd5d-2bbd-47ef-91b3-d59be2a596fb">
    <when expression="#[sizeof(payload) > 0]">
      <set-variable value="{message.element.position}"
        doc:name="Element Position"
        doc:id="a16c853b-d265-4e94-b2a0-81cbfec3653d"
        variableName="pos" />

      <choice doc:name="Choice"
        doc:id="7904f82a-21e1-41bd-b84a-18982d6b1af1">
        <when
          expression="#[vars.pos >= 0 and
            sizeof(payload) > vars.pos]">

          <set-variable
            value="#[output application/json ---
              payload[vars.pos as Number]]"
            doc:name="Get and Set Element"
            doc:id="321c6b2a-73b8-4088-83e2-
              58669f589d3c"
            variableName="elementValue" />
          <set-variable
            value="#[vars.elementValue.url
              splitBy ("/")]'"
            doc:name="Message Status Link
              Reference Value Split By '/'"
            doc:id="0fafd3df-8398-4c5f-b2f6-
              c91b0ce62469" variableName="link" />

          <set-variable value="#[vars.link[-1]]"
            doc:name="Message Id"
            doc:id="8f21b1cb-dae2-4ba0-944f-
              89dee07dd600"

```

```

567aee3680d0"
system-purchase-order-message-content
Purchase Order Message Content"
2330312112c5"
ref="Medius_Connector_Mule4_Config"
    messageStatusUriParam="#[vars.messageStatus]"
    messageIdUriParam="#[vars.messageId]" />
68d8dcc5362d"
0066ceb4c8f8"
736498692a87"
ERP-and-confirmation-flow" />
    </when>
    </choice>
  </when>
</choice>
</sub-flow>
<sub-flow
  name="send-purchase-order-to-ERP-and-confirmation-flow"
  doc:id="01722839-5bd3-4e66-a70b-b02a339822f5">
  <try doc:name="Try - Block for Sending Purchase Order to ERP ">
    <set-variable
      value="#[output application/json --- true as Boolean]"
      doc:name="Success Flag"
      doc:id="5273cd95-c972-4262-9d89-05a054dab7f0"
      variableName="isSendToERPSuccessfull" />
    <error-handler>
      <on-error-propagate enableNotifications="true"
        logException="true" doc:name="On Error Propagate"
        variableName="messageId" />
      <set-variable value="#[vars.link[-2]]"
        doc:name="Message Status"
        doc:id="50adfc2a-5a14-484e-8be6-
        variableName="messageStatus" />
      <mule4-medijs-connector:get-external-
        doc:name="Get External System
        doc:id="00264b57-f9ca-4f8f-a4fa-
        config-
        <logger level="INFO" doc:name="Logger"
          doc:id="47d58b68-f8f6-4bdd-9cf8-
          message="#[payload]" />
        <set-variable value="#[payload]"
          doc:name="Message Content"
          doc:id="01ad44e7-d630-49a2-802e-
          variableName="messageContent" />
        <flow-ref doc:name="Flow Reference"
          doc:id="274e7df8-b187-4c0d-ac65-
          name="send-purchase-order-to-

```

```

type="ANY">
23bc66402c20" message="#[error]" />
c4864b1196ad">
output application/java
---
write(error, "application/json"))]></ee:set-payload>
070f021db433"
a48e06990a0b"
214619b488dc"
ref="Medius_Connector_Mule4_Config">
procurement-error-body><![CDATA[#[%dw 2.0
output application/json
---
{
  "Errors": [
    {
      "Message": vars.errorRes
    }
  ],
  "CorrelationKey": vars.correlationKey
}]]]></mule4-medijs-connector:post-a-procurement-error-body>
b3f8bb32956c"
doc:id="9b50d2e0-ebb6-46a4-be1a-00bb533af964"
<logger level="INFO" doc:name="Error"
  doc:id="17eec6d-9eba-411c-8d01-
<ee:transform doc:name="Transform Message"
  doc:id="a665572c-ff77-48a3-bc36-
<ee:message>
  <ee:set-payload><![CDATA[%dw 2.0
</ee:message>
</ee:transform>
<set-variable value="#[payload]" doc:name="Error"
  doc:id="9a3e0892-d54d-4038-a21e-
  variableName="errorRes" />
<logger level="INFO" doc:name="Error Message"
  doc:id="6e75904d-c964-4069-937c-
  message="#[vars.errorRes]" />
<mule4-medijs-connector:post-a-procurement-error
  doc:name="Post A Procurement Error"
  doc:id="f7f21947-2a6d-4ef5-9a84-
  config-
<mule4-medijs-connector:post-a-
</mule4-medijs-connector:post-a-procurement-error>
<logger level="INFO" doc:name="Response Headers"
  doc:id="a452823f-d2b7-438c-b859-
  message="#[attributes]" />
</on-error-propagate>

```

```

        </error-handler>
    </try>
    <choice doc:name="Choice"
        doc:id="7ea8a349-191f-488e-9087-eb6193277129">
        <when expression="#[vars.isSendToERPSuccessfull == true]">
            <logger level="INFO" doc:name="Logger"
                doc:id="27803519-b5aa-4d81-95fc-9674cc444076"
                message="Send Purchase Order to ERP is successfully
done." />

            <ee:transform doc:name="Transform Message"
                doc:id="a0fec6c1-a9fe-482c-adcc-83e5d2e3b295">
                <ee:message>
                    <ee:set-payload><![CDATA[%dw 2.0
output application/json
---
{
    OrderNumber: vars.messageContent.orderNumber,
    Lines: vars.messageContent.lines map {
        LineNumber: $.lineNumber
    }
}]]></ee:set-payload>

                </ee:message>
            </ee:transform>
            <logger level="INFO" doc:name="Logger"
                doc:id="d63e2d78-fafe-4ade-afc7-4921fbf956c2"
                message="#[payload]" />

            <mule4-medijs-connector:purchase-order-to-erp-or-goods-
receipt-note-confirmation
                doc:name="Purchase Order To Erp Or Goods Receipt Note
Confirmation"
                doc:id="be116d31-b079-4028-a172-342dc7725934"
                config-ref="Medius_Connector_Mule4_Config">
                <mule4-medijs-connector:purchase-order-to-erp-or-
goods-receipt-note-confirmation-body><![CDATA[#[%dw 2.0
output application/json
---
{
    CorrelationKey: vars.correlationKey,
    PurchaseOrderConfirmation: {
        OrderNumber: payload.OrderNumber,
        Lines: payload.Lines
    }
}]]></mule4-medijs-connector:purchase-order-to-erp-or-goods-receipt-note-confirmation-body>
            </mule4-medijs-connector:purchase-order-to-erp-or-goods-
receipt-note-confirmation>

            <logger level="INFO" doc:name="Logger"
                doc:id="fa7861d4-4f2a-4a03-94ae-837ec6bc6508"
                message="#[attributes]" />

            <choice doc:name="Choice"

```

|                                                           |                                                     |
|-----------------------------------------------------------|-----------------------------------------------------|
|                                                           | doc:id="cabd0d5d-a436-41c0-92bb-5ef521b68273">      |
|                                                           | <when expression="#[attributes.statusCode == 201]"> |
| Success"                                                  | <logger level="INFO" doc:name="Confirmation         |
| 113ffee7dc7e"                                             | doc:id="bc8f15aa-150f-4aa9-b8bb-                    |
| Goods Receipt Note Confirmation is done successfully." /> | message="Purchase order to ERP or                   |
| splitBy ("/")'                                            | <set-variable                                       |
| Split By /"                                               | value='#[attributes.headers.location                |
| cb4d216e122c"                                             | doc:name="Location Header Value                     |
|                                                           | doc:id="3a761c35-2016-4a8e-8c4a-                    |
|                                                           | variableName="location" />                          |
|                                                           | <set-variable value="#[vars.location[-1]]"          |
|                                                           | doc:name="Message Id"                               |
| ffb7f56d060e"                                             | doc:id="658b1f72-7eae-49d4-b2f4-                    |
|                                                           | variableName="messageld" />                         |
| system-purchase-order-message-status                      | <mule4-medijs-connector:get-external-               |
| Purchase Order Message Status"                            | doc:name="Get External System                       |
| 77a049627a92"                                             | doc:id="cbadc3ed-d033-416b-93ee-                    |
| ref="Medius_Connector_Mule4_Config"                       | config-                                             |
| messageldUriParam="#[vars.messageld]" />                  |                                                     |
| Headers"                                                  | <logger level="INFO" doc:name="Response             |
| f5de85a21172"                                             | doc:id="b046afad-7c83-4c6d-a5ad-                    |
|                                                           | message="#[attributes]" />                          |
| a8defe3a72de"                                             | <logger level="INFO" doc:name="Response"            |
|                                                           | doc:id="1004f495-4c8b-4a1c-b856-                    |
| 62162d873b17">                                            | message="#[payload]" />                             |
| expression="#[attributes.statusCode == 200]">             | <choice doc:name="Choice"                           |
| value="#[payload.status]"                                 | doc:id="39d9ad76-6842-45a8-bab1-                    |
| doc:name="Message Status"                                 | <when                                               |
|                                                           | <set-variable                                       |

```

doc:id="de970ff3-d0b9-4b00-bc71-87f6ba86aeea"

variableName="msgStatus" />

connector:get-external-system-purchase-order-message-content
External System Purchase Order Message Content"

doc:id="43ce0c08-af28-4f13-a758-2d368478d4ba"
ref="Medius_Connector_Mule4_Config"

messageIdUriParam="#[vars.messageId]"

messageStatusUriParam="#[vars.msgStatus]" />

</when>
</choice>
</when>
</choice>
</when>
<otherwise>
  <try doc:name="Try - Block to Raise Error and Log in Medius"
    doc:id="b7eee9fd-f186-4ba2-a374-1e8be9da2fb0">
    <raise-error doc:name="Raise error"
      doc:id="d1594f2c-d9bd-4778-84c6-
3de5411182ff" type="ANY"
      description="Send Purchase Order to ERP is
failed." />
    <error-handler>
      <on-error-propagate
        logException="true" doc:name="On
Error Propagate"
        doc:id="b3504dcc-11ba-4919-b73a-
9cdf86bab691" type="ANY">
        <logger level="INFO"
          doc:id="e1ff5a20-9c2f-
4c5d-9fe7-524d9f530ebf" message="#[error]" />
        <set-variable
          doc:name="Error"
          variableName="errorRes"
        />
        <logger level="INFO"
          doc:id="be941fef-b7b4-
421c-8789-c45d91b6c1a1"

```

```

        message="#[vars.errorRes]" />
    procurement-error
    Procurement Error"
    47d6-a58b-69cc4e8aa4fc"
    ref="Medius_Connector_Mule4_Config">
connector:post-a-procurement-error-body><![CDATA[#[%dw 2.0
output application/json
---
{
  "Errors": [
    {
      "Message": vars.errorRes
    }
  ],
  "CorrelationKey": vars.correlationKey
}]]]></mule4-medi-us-connector:post-a-procurement-error-body>
procurement-error>
doc:name="Response Headers"
4992-beaf-f7f9f37d09da"
/>

        </on-error-propagate>
    </error-handler>
  </try>
</otherwise>
</choice>
</sub-flow>
</mule>
<mule4-medi-us-connector:post-a-
doc:name="Post      A
doc:id="1845b47a-c3d7-
config-
<mule4-medi-us-
</mule4-medi-us-connector:post-a-
<logger      level="INFO"
doc:id="cb264fad-e5d2-
message="#[attributes]"

```

## Supplier Invoices Post Payment Information Messages and Search, Fetch, Update Status

This use case demonstrates the usage of various Supplier Invoice operations for example Search, Fetch Message Status, Fetch Message Content, Update Specific as well as Bulk Update Messages status and shows example of Posting Payment Information Message and then fetching it based on selection of flow to execute.

- In Anypoint Studio, click **File > New > Mule Project**, name the project **mule-medi-us-connector-supplier-invoice-demo** and click Finish.
- In the search field, type **Http** and drag the **HTTP Listener** to the canvas.
- Click the **HTTP Listener**, click the green plus sign to the right of Connector Configuration. On the next screen, add the host and port, click OK
- Provide **Path** to HTTP Listener (Example: **/supplier-invoice**)
- On the Mule Palette, search for **Set Variable** and drag it onto the canvas. Set value of variable from **mule-application.properties** file key “**selected.option**” as **\${selected.option}** and give name of variable as **selectedOpt** and display name as **Selected Option**.
- There are two options available for execution of two different flows. This option user can change from **mule-application.properties** file.
  - **MSG\_SEARCH\_FETCH\_UPDATE** - execute Search, Fetch Status, Fetch Content, Update Status and Bulk Update Status operations related flow
  - **PAYMENT\_INFO** - it will post payment information messages and then based on location header value it will fetch payment information message status.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- There will be two **When** blocks for added **Choice**. Those will be as follows.
- Add below condition as expression in **Choice** to check if user wants to execute Search, Fetch Status, Fetch Content, Update Status and Bulk Update Status operations related flow.

```
vars.selectedOpt == 'MSG_SEARCH_FETCH_UPDATE'
```

- Add **Flow Reference** to call another sub flow '**supplier-invoice-message-flow**' to Search, Fetch Status, Fetch Content, Update Status and Bulk Update Status operations related flow.
- Add below condition as expression in **Choice** to check if the user wants to post payment information messages and then based on location header value it will fetch payment information message status operation related flow.

```
vars.selectedOpt == 'PAYMENT_INFO'
```

- Add **Flow Reference** to call another sub flow '**payment-information-message-flow**' flow.
- Create a new sub flow **supplier-invoice-message-flow**.
- Add **Set Variable** and set variable name as **searchStatus** and value as **\${search.status}** key from **mule-application.properties** file and display name as **Search Status**.
- Add **Search Invoices** Medius operation from **Mule Palette** to canvas and provide **status** as expression **upper(vars.searchStatus)**. This operation returns all messages that match the

parameters sent in. The mandatory parameter is the status parameter, besides that the other parameters are sent in as query string parameters to narrow down the result.

- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if payload has any record.

`sizeof(payload) > 0`

- Add **Set Variable** and set variable name as **pos** and value as `${message.element.position}` key from **mule-application.properties** file and display name as **Element Position**. It will say which record user wants to process further based on position. For the first record its value will be 0, for the second record it will have 1 and so on.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if selected position is greater than or equal to zero and size of payload has record for that particular position.

`vars.pos >= 0 and sizeof(payload) > vars.pos`

- Add **Set Variable** and set variable name as **elementValue** and value as expression **output application/json --- payload[vars.pos as Number]** and display name as **Get and Set Element Value**. It will fetch the element from a particular position.
- Add **Set Variable** and set variable name as **link** and value as expression **vars.elementValue.url splitBy ("/")** and display name as **Message Status Link Reference Value Split By /**. It will be useful to set a message Id and message status.
- Add **Set Variable** and set variable name as **messageId** and value as expression **vars.link[-1]** and display name as **Message Id** here -1 means last element from array and it is messageId.
- Add **Set Variable** and set variable name as **messageStatus** and value as expression **vars.link[-2]** and display name as **Message Status** here -2 means the second last element from array and it is messageStatus.
- Search **Get Supplier Invoice Message Status** Medius operation from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId**. The operation with message id as input will return a status and an effective link to the real supplier invoice content. The main idea with this method is to be able to check status for a given message and to be able to find where a message is in terms of statuses.
- Add **Logger** to log payload response and set message as expression **payload**
- Search **Get Supplier Invoice Message Content** from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId** and Message Status as expression **vars.messageStatus**. This operation returns the supplier invoice content from a message that matches the status and the id. If the

message id is in another status, nothing will be returned. To find a message, use the other GET method that only provides message id as input.

- Add **Set Variable** and set variable name as **corKey** and value as expression **payload.CorrelationKey** and display name as **Set Correlation Key**
- Add **Logger** to log response and set message as expression **payload**
- Add **Set Variable** and set variable name as **newStatus** and value as **\${new.update.status}** from mule properties file and display name as **New Update Status**. This indicates a new update status for a given record. It will be useful in further operations.
- Search **Update Supplier Invoice Status** from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.messageId** and Message Status as expression **vars.messageStatus** and provide **body** parameter as expression as follows:

```
%dw 2.0
output application/json
---
{
    "NewStatus": vars.newStatus
}
```

- This operation updates the status for a supplier invoice message when processing it.
- Add **Logger** to log response and set message as expression **payload**
- Search **Bulk Update Invoice Status** from **Mule Palette** and drag to canvas and provide Status as expression **upper(vars.newStatus)** and Correlation Key Filter as expression **vars.corKey** and provide **body** parameter as expression as follows:

```
%dw 2.0
output application/json
---
{
    "NewStatus": vars.messageStatus
}
```

- This operation updates the status for all messages that meet the filter parameters.
- Add **Logger** to log payload and set message as expression **payload**
- Create a new sub flow **payment-information-message-flow**.
- Add **Set Variable** and set variable name as **companyId** and value as **\${company.id}** key from **mule-application.properties** file and display name as **Company Id**.

- Add **Set Variable** and set variable name as **supplierId** and value as **\${supplier.id}** key from **mule-application.properties** file. and display name as **Supplier Id**.
- Add **Set Variable** and set variable name as **invoiceNumber** and value as **\${invoice.number}** key from **mule-application.properties** file. and display name as **Invoice Number**.
- Add **Post Payment Information Message** Medius operation from **Mule Palette** to canvas and provide **body** as expression :

```
%dw 2.0
output application/json
---
{
  "Invoice": {
    "Date": "2023-02-09T05:55:14.358Z",
    "SupplierId": vars.supplierId,
    "CompanyId": vars.companyId,
    "InvoiceNumber": vars.invoiceNumber
  },
  "PaymentLine": {
    "Date": "2023-02-09T05:55:14.358Z",
    "Amount": 100,
    "Status": "Final",
    "Method": "Bank",
    "Identifier": "500050500",
    "ExternalIdentifier": "Bank;500050500;1",
    "AdditionalInformation": "Final payment for invoice"
  }
}
```

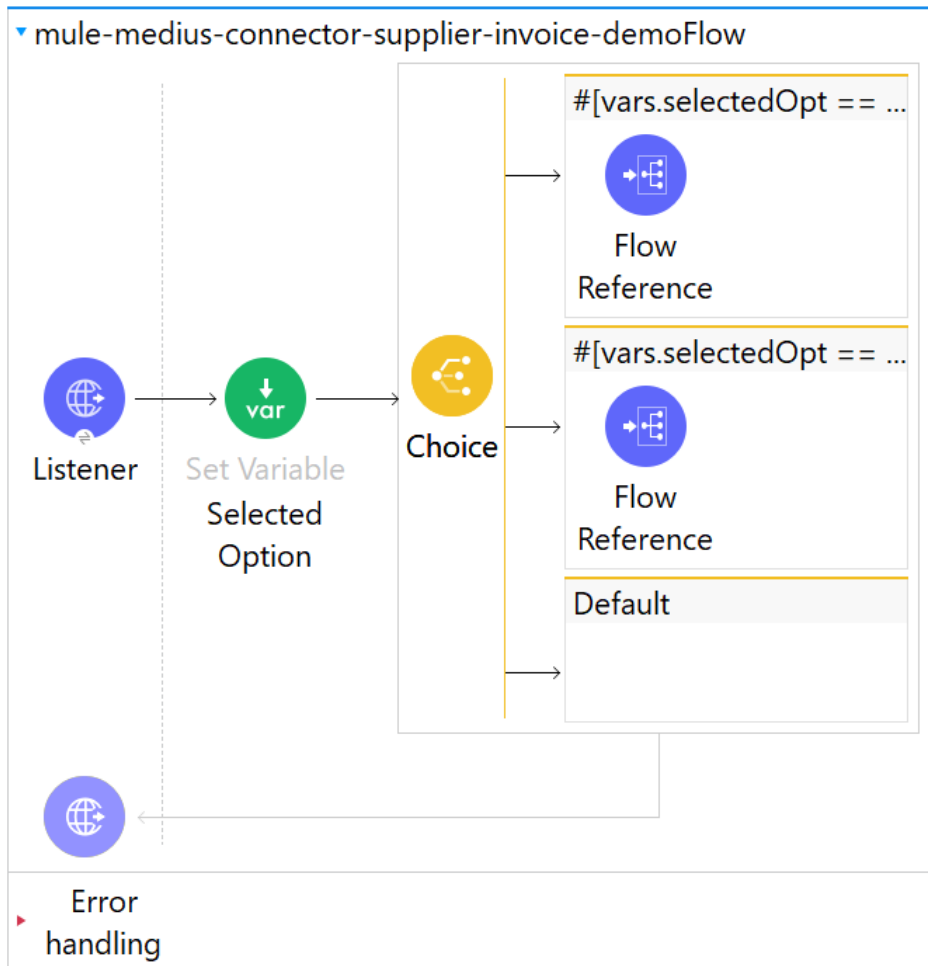
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if response has location header.

```
attributes.headers.location != null
```

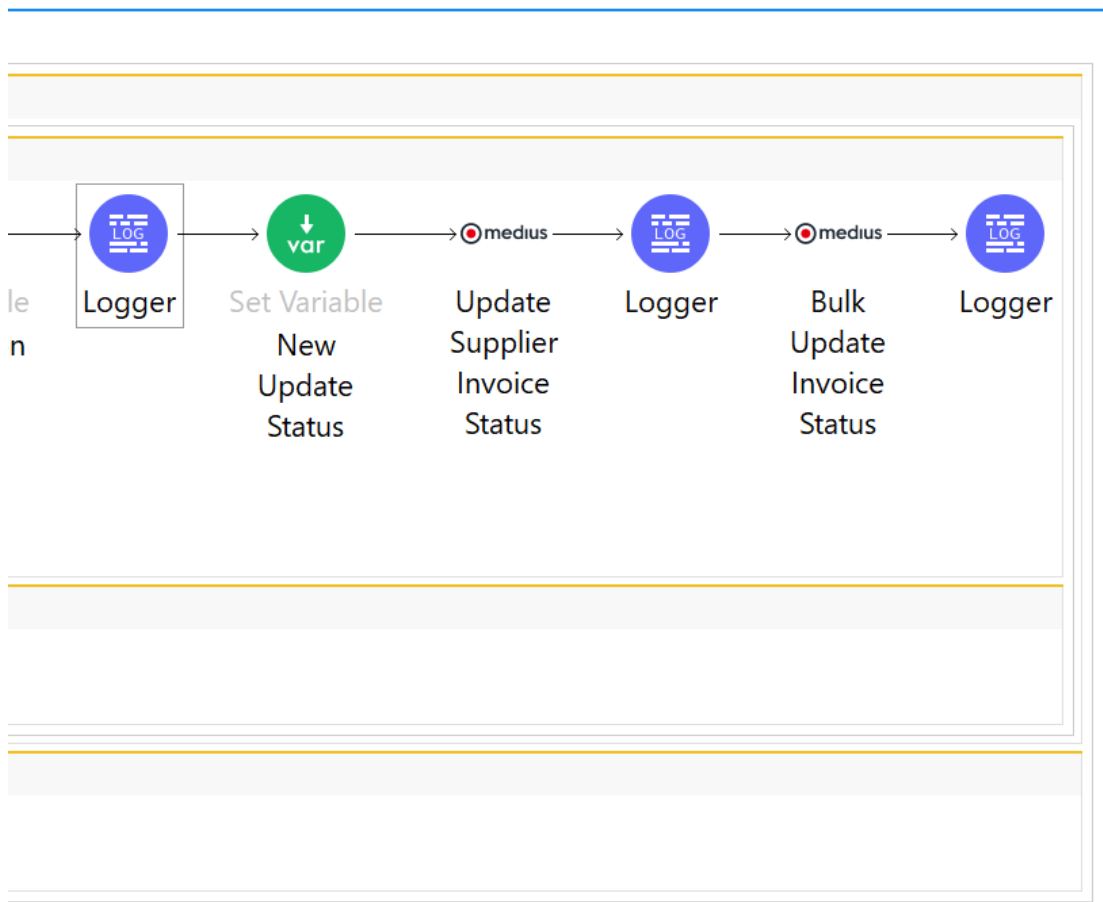
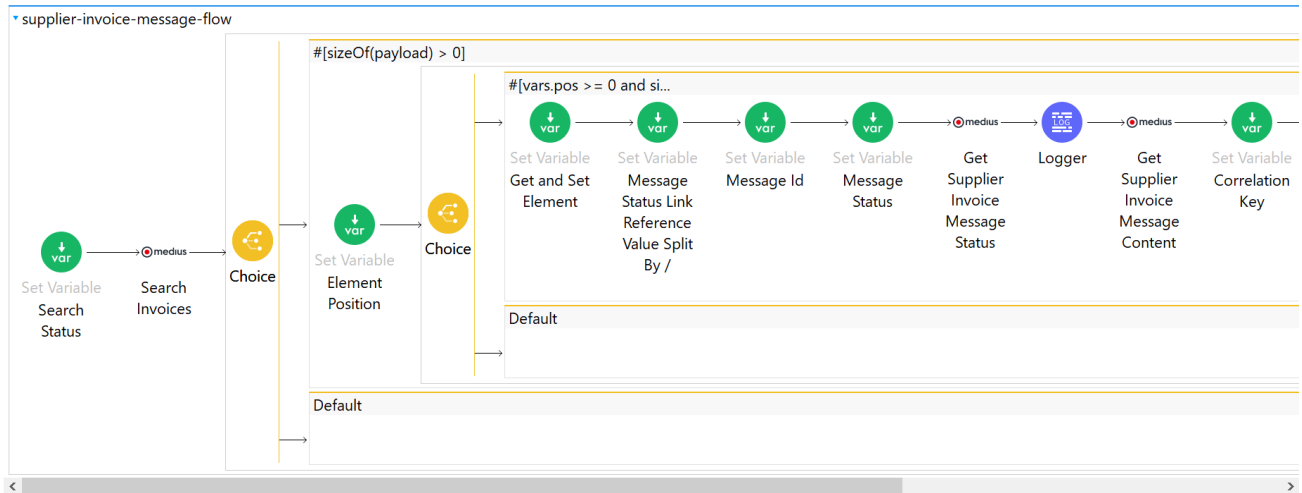
- Add **Set Variable** and set variable name as **link** and value as expression **attributes.headers.location splitBy ("/")** and display name as **Link Reference Value Split By /**. It will be useful to set a message Id.
- Add **Set Variable** and set variable name as **msgId** and value as expression **vars.link[-1]** and display name as **Message Id**. Here -1 means the last element from the array and it is messageId.

- Search **Get Payment Information Message Status** from **Mule Palette** and drag to canvas and provide Message Id as expression **vars.msgId**.
- After creating the flow, right-click the project and click **Run As > Mule Application**.
- Once the application is deployed, use **HTTP Listener's Path** to execute the Mule Flow.

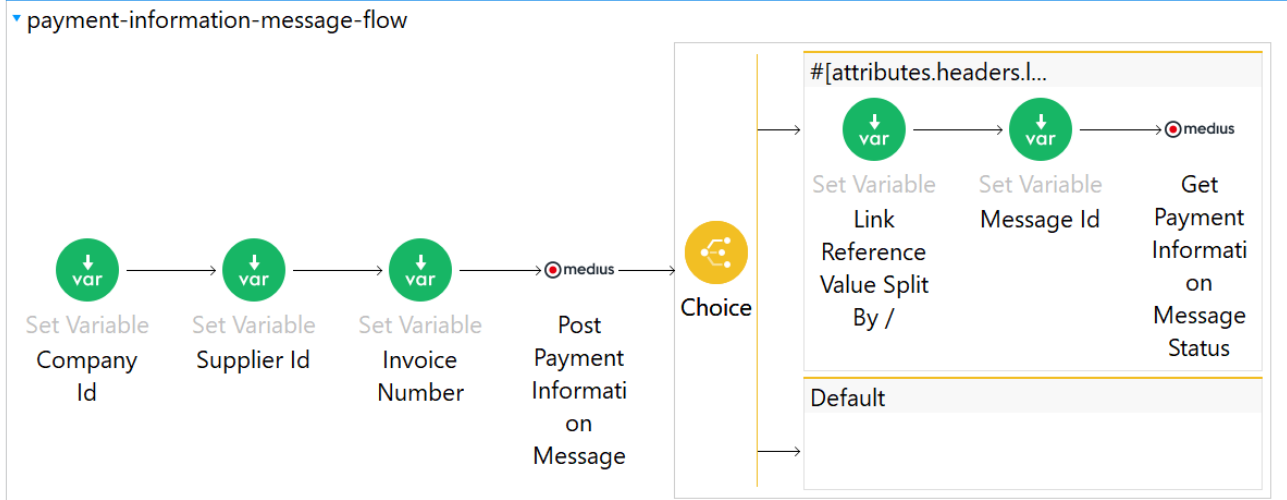
#### mule-mediuis-connector-supplier-invoice-demoFlow



#### supplier-invoice-message-flow



payment-information-message-flow



**Use Case Configuration XML** : Paste this XML code into Anypoint Studio to experiment with the flow described above.

```

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

<mule
  xmlns:mule4-mediuss-connector="http://www.mulesoft.org/schema/mule/mule4-mediuss-
connector"
  xmlns:http="http://www.mulesoft.org/schema/mule/http"
  xmlns="http://www.mulesoft.org/schema/mule/core"
  xmlns:doc="http://www.mulesoft.org/schema/mule/documentation"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.mulesoft.org/schema/mule/core
http://www.mulesoft.org/schema/mule/core/current/mule.xsd
http://www.mulesoft.org/schema/mule/http
http://www.mulesoft.org/schema/mule/http/current/mule-http.xsd
http://www.mulesoft.org/schema/mule/mule4-mediuss-connector
http://www.mulesoft.org/schema/mule/mule4-mediuss-connector/current/mule-mule4-mediuss-
connector.xsd">

  <configuration-properties
    file="mule-application.properties" />

  <http:listener-config name="HTTP_Listener_config"
    doc:name="HTTP Listener config"
    doc:id="09f5cca2-7cd3-437e-a268-5ee0c7824a3d">
    <http:listener-connection host="0.0.0.0"
      port="8081" />
  </http:listener-config>

  <mule4-mediuss-connector:config
    name="Mediuss_Connector_Mule4_Config"

```

```

doc:name="Medius Connector Mule4 Config"
doc:id="1adba4ea-02c5-4d97-a897-a0b14afe87be">
<mule4-medijs-connector:medijsflowcloudauth-connection
    baseUri="{medijs.base.url}">
    <mule4-medijs-connector:oauth-client-credentials
        clientId="{medijs.client.id}" tokenUrl="{medijs.token.url}"
        clientSecret="{medijs.client.secret}" scopes="{medijs.scopes}"
    />
</mule4-medijs-connector:medijsflowcloudauth-connection>
</mule4-medijs-connector:config>

<flow name="mule-medijs-connector-supplier-invoice-demoFlow"
doc:id="44b03b34-9f23-49a9-b06c-359d8bc8bac2">
<http:listener doc:name="Listener"
    doc:id="a25dfd77-e60b-4e52-9d74-97b820122e98"
    config-ref="HTTP_Listener_config" path="/supplier-invoice">
</http:listener>

<set-variable value="{selected.supplier.invoice.option}"
    doc:name="Selected Option"
    doc:id="4525e3bb-2cf1-4812-a1f8-5866676e17ed"
    variableName="selectedOpt" />

<choice doc:name="Choice"
    doc:id="29e83357-896a-4227-9944-3870ba99a548">
    <when
        expression="#[vars.selectedOpt
'MSG_SEARCH_FETCH_UPDATE']">

        <flow-ref doc:name="Flow Reference"
            doc:id="60b14796-6b2a-4d16-9227-0d4247713d63"
            name="supplier-invoice-message-flow" />
    </when>
    <when expression="#[vars.selectedOpt == 'PAYMENT_INFO']">

        <flow-ref doc:name="Flow Reference"
            doc:id="6252c0ca-de42-4d3f-8bbf-2d69137f8edd"
            name="payment-information-message-flow" />
    </when>
</choice>
</flow>

<sub-flow name="supplier-invoice-message-flow"
doc:id="c2513693-0ed2-4527-a88e-4b06a9e82c6a">

    <set-variable value="{search.status}" doc:name="Search Status" doc:id="48737221-
d218-4270-8a62-8d9cad806fe3" variableName="searchStatus" />
    <mule4-medijs-connector:search-invoices
        doc:name="Search Invoices"
        doc:id="46584bd3-2753-49cb-bfc0-1f9cb5171a0d"

```

```

config-ref="Medius_Connector_Mule4_Config"
statusUriParam="#[upper(vars.searchStatus)]" />

<choice doc:name="Choice"
  doc:id="1ba099a3-f30b-4745-89e5-b5770cd0f565">
  <when expression="#[sizeOf(payload) > 0]">
    <set-variable value="{message.element.position}"
      doc:name="Element Position"
      doc:id="26aa86ea-d211-42f3-8ddf-4b25661729fb"
variableName="pos" />

    <choice doc:name="Choice"
      doc:id="460d8681-ee4b-436a-ab28-af847c6fbfd3">
        <when
          expression="#[vars.pos >= 0 and
sizeOf(payload) > vars.pos]">

          <set-variable
            value="#[output application/json ---
payload[vars.pos as Number]]"
            doc:name="Get and Set Element"
            variableName="elementValue" />
          <set-variable
            value="#[vars.elementValue.url
splitBy ("/")]'"
            doc:name="Message Status Link
Reference Value Split By /"
            doc:id="98df4332-34db-458d-86a9-
4ab806da04ed" variableName="link" />

          <set-variable value="#[vars.link[-1]]"
            doc:name="Message Id"
            doc:id="a23e0f80-10de-4f70-9922-
a6e2ebbc3cd1"
            variableName="messageId" />
          <set-variable value="#[vars.link[-2]]"
            doc:name="Message Status"
            doc:id="5b4607f9-cac5-4f63-ba9e-
63cf5258952a"
            variableName="messageStatus" />
          <mule4-medijs-connector:get-supplier-
            doc:name="Get Supplier Invoice
Message Status"
            doc:id="c3ce4816-30dd-4e10-bf15-
28bf51f78194"
            config-
ref="Medius_Connector_Mule4_Config"

            messageIdUriParam="#[vars.messageId]" />

          <logger level="INFO" doc:name="Logger"

```

|                                                                   |                                                                                                                                          |
|-------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------|
| 8a4cfd22ef4b"                                                     | doc:id="dbb17b81-8e7f-41fe-a854-                                                                                                         |
| invoice-message-content                                           | message="#[payload]" /><br><mule4-medijs-connector:get-supplier-                                                                         |
| Message Content"                                                  | doc:name="Get Supplier Invoice                                                                                                           |
| df48ff13647d"                                                     | doc:id="7dc4b6ff-c51f-47cb-a164-                                                                                                         |
| ref="Medius_Connector_Mule4_Config"                               | config-                                                                                                                                  |
| messageStatusUriParam="#[vars.messageStatus]"                     |                                                                                                                                          |
| messageIdUriParam="#[vars.messageId]" />                          |                                                                                                                                          |
| value="#[payload.CorrelationKey]"                                 | <set-variable                                                                                                                            |
| 1950d6f56e5f"                                                     | doc:name="Correlation Key"<br>doc:id="0ef7dec5-afb9-4c3f-8756-                                                                           |
| 79a9460ba314"                                                     | variableName="corKey" /><br><logger level="INFO" doc:name="Logger"<br>doc:id="84a89c12-247d-4d19-a4f4-                                   |
| 8970cd573981"                                                     | message="#[payload]" /><br><set-variable value="{new.update.status}"<br>doc:name="New Update Status"<br>doc:id="86f6bdeb-5f0d-4a3a-a898- |
| invoice-status                                                    | variableName="newStatus" /><br><mule4-medijs-connector:update-supplier-                                                                  |
| Status"                                                           | doc:name="Update Supplier Invoice                                                                                                        |
| e89874159d28"                                                     | doc:id="cad74d82-be7e-490e-a774-                                                                                                         |
| ref="Medius_Connector_Mule4_Config"                               | config-                                                                                                                                  |
| messageIdUriParam="#[vars.messageId]"                             |                                                                                                                                          |
| messageStatusUriParam="#[vars.messageStatus]">                    |                                                                                                                                          |
| supplier-invoice-status-body><![CDATA[#{%dw 2.0                   | <mule4-medijs-connector:update-                                                                                                          |
| output application/json                                           |                                                                                                                                          |
| ---                                                               |                                                                                                                                          |
| {                                                                 |                                                                                                                                          |
| "NewStatus": vars.newStatus                                       |                                                                                                                                          |
| }}]></mule4-medijs-connector:update-supplier-invoice-status-body> |                                                                                                                                          |

|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |                                                                            |
|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------|
| invoice-status>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | </mule4-medius-connector:update-supplier-                                  |
| 80ed3125d18c"                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | <logger level="INFO" doc:name="Logger"<br>doc:id="d686f4bb-986e-4661-ad8f- |
| invoice-status                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | message="#[payload]" /><br><mule4-medius-connector:bulk-update-            |
| Status"                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | doc:name="Bulk Update Invoice                                              |
| 03f2d5686f70"                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | doc:id="5c87e24f-8036-4a22-b94d-                                           |
| ref="Medius_Connector_Mule4_Config"                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   | config-                                                                    |
| statusUriParam="#[upper(vars.newStatus)]"                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                                                                            |
| correlationKeyFilterQueryParam="#[vars.corKey]">                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | <mule4-medius-connector:bulk-                                              |
| update-invoice-status-body><![CDATA[#[%dw 2.0<br>output application/json<br>---<br>{<br>"NewStatus": vars.messageStatus<br>}]]]></mule4-medius-connector:bulk-update-invoice-status-body>                                                                                                                                                                                                                                                                                                                                                                             |                                                                            |
| invoice-status>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | </mule4-medius-connector:bulk-update-                                      |
| 758619bd1570"                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | <logger level="INFO" doc:name="Logger"<br>doc:id="545d0c54-3aa2-4b4b-a966- |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | message="#[payload]" />                                                    |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | </when>                                                                    |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | </choice>                                                                  |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | </when>                                                                    |
|                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       | </choice>                                                                  |
| </sub-flow>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                                                                            |
| <sub-flow name="payment-information-message-flow"<br>doc:id="ce4af12f-56b7-4595-90f3-ddd7c9fce758"><br><set-variable value="\$\${company.id}" doc:name="Company Id"<br>doc:id="6c3b4cd2-62e5-48ba-9830-acf1cf202455"<br>variableName="companyId" /><br><set-variable value="\$\${supplier.id}"<br>doc:name="Supplier Id" doc:id="cac6cf83-9028-4969-b7f8-49dcbb827bc8"<br>variableName="supplierId" /><br><set-variable value="\$\${invoice.number}"<br>doc:name="Invoice Number"<br>doc:id="e0c27c23-8ceb-4fed-8c0a-ba9c7a7a8220"<br>variableName="invoiceNumber" /> |                                                                            |

```

<mule4-medius-connector:post-payment-information-message
  doc:name="Post Payment Information Message"
  doc:id="bdbfd470-8f3c-45cf-943e-7f0b716e150b"
  config-ref="Medius_Connector_Mule4_Config">
  <mule4-medius-connector:post-payment-information-message-
body><![CDATA[#[%dw 2.0
output application/json
---
{
  "Invoice": {
    "Date": "2023-02-09T05:55:14.358Z",
    "SupplierId": vars.supplierId,
    "CompanyId": vars.companyId,
    "InvoiceNumber": vars.invoiceNumber
  },
  "PaymentLine": {
    "Date": "2023-02-09T05:55:14.358Z",
    "Amount": 100,
    "Status": "Final",
    "Method": "Bank",
    "Identifier": "500050500",
    "ExternalIdentifier": "Bank;500050500;1",
    "AdditionalInformation": "Final payment for invoice"
  }
}]]]></mule4-medius-connector:post-payment-information-message-body>
  </mule4-medius-connector:post-payment-information-message>
  <choice doc:name="Choice"
    doc:id="68b8d324-835e-436e-8c8e-14e944cdc032">
    <when expression="#[attributes.headers.location != null]">
      <set-variable
        value='#[attributes.headers.location splitBy ("/")]'
        doc:name="Link Reference Value Split By /"
        doc:id="461bb361-8148-425f-9956-63ce836f5cbd"
variableName="link" />
      <set-variable value="#[vars.link[-1]]" doc:name="Message Id"
doc:id="49b074b9-67ef-4faa-aaa2-d899b7cb6da2" variableName="msgId"/>
      <mule4-medius-connector:get-payment-information-message-
status doc:name="Get Payment Information Message Status" doc:id="13adfd31-7f1f-4dc9-bfc8-
ca17b1dd458d" config-ref="Medius_Connector_Mule4_Config" messageIdUriParam="#[vars.msgId]"/>
      </when>
    </choice>
  </sub-flow>
</mule>

```

## Master Data Import

This use case demonstrates import of master data based on selection of flow to execute.

- In Anypoint Studio, click **File > New > Mule Project**, name the project **mule-medi-us-connector-master-data-import-demo** and click Finish.
- In the search field, type **Http** and drag the **HTTP Listener** to the canvas.
- Click the **HTTP Listener**, clickz the green plus sign to the right of Connector Configuration. On the next screen, add the host and port, click OK
- Provide **Path** to HTTP Listener (Example: **/master-data-import**)
- On the Mule Palette, search for **Set Variable** and drag it onto the canvas. Set value of variable from **mule-application.properties** file key “**selected.option**” as **\${selected.import.option}** and give name of variable as **selectedImportOption** and display name as **Selected Import Option**.
- On the Mule Palette, search for **Set Variable** and drag it onto the canvas. Set value of variable from **mule-application.properties** file key “**selected.option**” as **\${erp.source.id}** and give name of variable as **erpSourceId** and display name as **ERP Source Id**.
- There are three options available for execution of three different flows. This option user can change from **mule-application.properties** file.
  - **RESTRICTION\_RULE** - allows the client to send restriction rules that will be imported by the data import queue. The Restriction rules have company prerequisites.
  - **DIMENSION\_VALUE** - allows the client to send dimension values that will be imported by the Medius master data import queue. The dimension values have company as a mandatory prerequisite.
  - **SUPPLIER** - allows the client to send suppliers that will be imported by the Medius master data import queue. The supplier has a payment term and company as a mandatory prerequisite. If additional dimensions are going to be used or Vat groups they need to be imported in advance.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- There will be three **When** blocks for added **Choice**. Those will be as follows.
- Add below condition as expression in **Choice** to check if user wants to execute restriction rule master data import operations related flow.

```
vars.selectedImportOption == 'RESTRICTION_RULE'
```

- Add **Flow Reference** to call another sub flow '**restriction-rule-flow**'.
- Add below condition as expression in **Choice** to check if the user wants to execute dimension value master data import.

```
vars.selectedImportOption == 'DIMENSION_VALUE'
```

- Add **Flow Reference** to call another sub flow '**dimension-value-flow**' flow.
- Add below condition as expression in **Choice** to check if the user wants to execute supplier master data import.

```
vars.selectedImportOption == 'SUPPLIER'
```

- Add **Flow Reference** to call another sub flow '**supplier-flow**' flow.
- After the above **Choice** block, add another **Choice** block.
- Add below condition as expression in **Choice** to check if selected import option is among supported three master data imports then get batch information using import id received as part of sub flow.

```
vars.selectedImportOption == 'RESTRICTION_RULE' or vars.selectedImportOption ==  
'DIMENSION_VALUE' or vars.selectedImportOption == 'SUPPLIER'
```

- Add **Set Variable** and set variable name as **location** and value as expression **attributes.headers.location splitBy ("/")** and display name as **Location Header Value Split By /**.
- Add **Set Variable** and set variable name as **importId** and value as expression **vars.location[-1]** and display name as **Import Id**. Here -1 means the last element from an array.
- Add **Logger** to log import id and set message as expression **vars.importId**.
- Add **Get Import Batch Job Status** Medius operation and provide master data type as expression **vars.selectedImportOption** and import id as expression **vars.importId**. This operation is used to get the status of the import job for any of the master data. When performing a POST, the link returned is pointing to this method. This method allows the client to orchestrate so that the client can wait until dependent master data is imported correctly before sending that data. In This operation importId is used what we get in Import Batch Job operation (Step 1) as part of response header named **location** by splitting header value with / and then taking the last element as importId.
- Create a new sub flow **restriction-rule-flow**.
- Add **Set Variable** and set variable name as **companyId** and value as **\${company.id}** key from **mule-application.properties** file and display name as **Company Id**.
- Add **Restriction Rule Import Batch** Medius operation from **Mule Palette** to canvas and provide **body** as expression :

```
%dw 2.0  
output application/json  
---  
[
```

```

{
  "ErpSourceId": vars.erpSourceId,
  "ExternalSystemId": "ExternalSystemIdBatchNo1",
  "CompanyId": vars.companyId,
  "IsActive": true,
  "RestrictionRules": [
    {
      "ControllingDimension": "DIMENSION1",
      "ControllingDimensionValues": [
        {
          "Is": "1150"
        }
      ],
      "RestrictedOrderedDimensionGroups": [
        {
          "Dimension": "DIMENSION2",
          "ChildRules": [
            {
              "Type": "forbidden",
              "DimensionValues": {
                "IsEmptyValue": true
              }
            }
          ]
        }
      ]
    },
    {
      "ControllingDimension": "DIMENSION1",
      "ControllingDimensionValues": [
        {
          "Is": "Q103"
        }
      ],
      "RestrictedOrderedDimensionGroups": [
        {
          "Dimension": "DIMENSION3",
          "ChildRules": [

```

```

    {
      "Type": "fixed",
      "DimensionValues": {
        "Is": "Q303"
      }
    }
  ]
},
{
  "ControllingDimension": "DIMENSION2",
  "ControllingDimensionValues": [
    {
      "IsAnyValue": true
    }
  ],
  "RestrictedOrderedDimensionGroups": [
    {
      "Dimension": "DIMENSION3",
      "ChildRules": [
        {
          "Type": "forbidden",
          "DimensionValues": {
            "IsEmptyValue": true
          }
        }
      ]
    }
  ]
},
{
  "ControllingDimension": "DIMENSION1",
  "ControllingDimensionValues": [
    {
      "Is": "1111"
    }
  ],

```

```

"RestrictedOrderedDimensionGroups": [
  {
    "Dimension": "DIMENSION2",
    "ChildRules": [
      {
        "Type": "forbidden",
        "DimensionValues": {
          "IsAnyValue": true
        }
      }
    ]
  },
  {
    "Dimension": "DIMENSION3",
    "ChildRules": [
      {
        "Type": "required",
        "DimensionValues": {
          "From": "22",
          "To": "24",
          "IncludeEdges": true
        }
      }
    ]
  }
],
{
  "ControllingDimension": "DIMENSION1",
  "ControllingDimensionValues": [
    {
      "Is": "1112"
    }
  ],
  "RestrictedOrderedDimensionGroups": [
    {
      "Dimension": "DIMENSION2",
      "ChildRules": [

```

```

    {
      "Type": "optional",
      "DimensionValues": {
        "IsEmptyValue": true
      }
    }
  ],
},
    {
      "Dimension": "DIMENSION3",
      "ChildRules": [
        {
          "Type": "fixed",
          "DimensionValues": {
            "Is": "22"
          }
        }
      ]
    }
  ]
}
],
},
{
  "ErpSourceId": vars.erpSourceId,
  "ExternalSystemId": "ExternalSystemIdBatchNo2",
  "CompanyId": vars.companyId,
  "IsActive": true,
  "RestrictionRules": [
    {
      "ControllingDimension": "DIMENSION1",
      "ControllingDimensionValues": [
        {
          "Is": "1120"
        }
      ],
    },
    "RestrictedOrderedDimensionGroups": [
      {

```

```

    "Dimension": "DIMENSION2",
    "ChildRules": [
      {
        "Type": "forbidden",
        "DimensionValues": {
          "IsEmptyValue": true
        }
      }
    ]
  }
]
}
]
}
]
}
]

```

- This operation allows the client to send restriction rules that will be imported by data import queue. The Restriction rules have company prerequisites.
- Create a new sub flow **dimension-value-flow**.
- Add **Set Variable** and set variable name as **externalSysIdOfCompany1** and value as **\${external.system.company.id};\${dimension.name};\${dimension.value.1}** key from **mule-application.properties** file and display name as **External System Id of Company For Dimesnion 1**.
- Add **Set Variable** and set variable name as **externalSysIdOfCompany2** and value as **\${external.system.company.id};\${dimension.name};\${dimension.value.2}** key from **mule-application.properties** file and display name as **External System Id of Company For Dimesnion 2**.
- Add **Set Variable** and set variable name as **company** and value as **\${company.id.for.dimension.value}** key from **mule-application.properties** file and display name as **Company Id**.
- Add **Set Variable** and set variable name as **dimensionName** and value as **\${dimension.name}** key from **mule-application.properties** file and display name as **Dimension Name**.
- Add **Set Variable** and set variable name as **dimensionVal1** and value as **\${dimension.value.1}** key from **mule-application.properties** file and display name as **Dimension Value 1**.
- Add **Set Variable** and set variable name as **dimensionVal2** and value as **\${dimension.value.2}** key from **mule-application.properties** file and display name as **Dimension Value 2**.
- Add **Dimension Value Import Batch** Medius operation from **Mule Palette** to canvas and provide **body** as expression :

```
%dw 2.0
```

output application/json

---

```
[
  {
    "ErpSourceId": vars.erpSourceId,
    "ExternalSystemId": vars.externalSysIdOfCompany1,
    "IsActive": true,
    "CompanyId": vars.company,
    "DimensionName": vars.dimensionName,
    "Description": "Internal purchases",
    "Value": vars.dimensionVal1,
    "TaxIndicator1": "",
    "TaxIndicator2": "",
    "IsFictive": true,
    "UsableInPurchase": true
  },
  {
    "ErpSourceId": vars.erpSourceId,
    "ExternalSystemId": vars.externalSysIdOfCompany2,
    "IsActive": true,
    "CompanyId": vars.company,
    "DimensionName": vars.dimensionName,
    "Description": "External purchases",
    "Value": vars.dimensionVal2,
    "TaxIndicator1": "",
    "TaxIndicator2": "",
    "IsFictive": true,
    "UsableInPurchase": true
  }
]
```

- This operation allows the client to send dimension values that will be imported by the Medius master data import queue. The dimension values have company as mandatory prerequisite.
- Create a new sub flow **supplier-flow**.
- Add **Set Variable** and set variable name as **externalSysId** and value as **\${external.system.company.id};\${supplier.id}** key from **mule-application.properties** file and display name as **External System Id of Company**.
- Add **Set Variable** and set variable name as **company** and value as **\${company.id.for.dimension.value}**

key from **mule-application.properties** file and display name as **Company Id**.

- Add **Set Variable** and set variable name as **supplierId** and value as **\${supplier.id}** key from **mule-application.properties** file and display name as **Supplier Id**.
- Add **Set Variable** and set variable name as **currencyCode** and value as **\${currency.code}** key from **mule-application.properties** file and display name as **Currency Code**.
- Add **Set Variable** and set variable name as **deliveryTerm** and value as **\${delivery.term}** key from **mule-application.properties** file and display name as **Delivery Term**.
- Add **Set Variable** and set variable name as **paymentTermId** and value as **\${external.system.company.id};\${payment.term.id}** key from **mule-application.properties** file and display name as **Payment Term Id**.
- Add **Set Variable** and set variable name as **ledgerDimension** and value as **\${external.system.company.id};\${ledger.dimension};\${ledger.dimension.value}** key from **mule-application.properties** file and display name as **Ledger Dimension**.
- Add **Set Variable** and set variable name as **taxDimension** and value as **\${external.system.company.id};\${tax.dimension};\${tax.dimension.value}** key from **mule-application.properties** file and display name as **Tax Dimension**.
- Add **Set Variable** and set variable name as **clearanceDimension** and value as **\${external.system.company.id};\${clearance.dimension};\${clearance.dimension.value}** key from **mule-application.properties** file and display name as **Clearance Dimension**.
- Add **Set Variable** and set variable name as **taxGroupCode** and value as **\${external.system.company.id};\${tax.group.code}** key from **mule-application.properties** file and display name as **Tax Group Code**.
- Add **Set Variable** and set variable name as **externalSysIdPaymentDetail** and value as **\${external.system.company.id};\${supplier.id};\${currency.code}** key from **mule-application.properties** file and display name as **External System Id of Company**.
- Add **Supplier Import Batch** Medius operation from **Mule Palette** to canvas and provide **body** as expression :

```
%dw 2.0
output application/json
---
[
    {
        "ErpSourceId": vars.erpSourceId,
        "ExternalSystemId": vars.externalSysId,
        "IsActive": true,
        "CompanyId": vars.company,
```

```

"CurrencyCode": vars.currencyCode,
"Identifiers": [
  {
    "IdentifierType": "FINANCIAL_SYSTEM",
    "IdentifierValue": vars.supplierId
  },
  {
    "IdentifierType": "BANK_ACCOUNT",
    "IdentifierValue": "123456789"
  },
  {
    "IdentifierType": "ORGANIZATION_NUMBER",
    "IdentifierValue": "99999999"
  }
],
"Name": "TestSupplier5",
"PaymentTerm": vars.paymentTermId,
"DeliveryTerm": vars.deliveryTerm,
"SupplierId": vars.supplierId,
"LedgerDimension1": vars.ledgerDimension,
"TaxDimension1": vars.taxDimension,
"Clearance1Dimension1": vars.clearanceDimension,
"TaxIndicator1": vars.taxGroupCode,
"TaxIndicator2": "",
"StreetAddress": "TestStreet",
"City": "TestCity",
"Zip": "0000",
"Country": "TestCountry",
"Telephone": "123456",
"Fax": "",
"Homepage": "",
"EmailAddress": "",
"State": "",
"IsProcurable": true,
"Prepayment": false,
"PaymentDetails": [
  {
    "ExternalSystemId": vars.externalSysIdPaymentDetail,

```

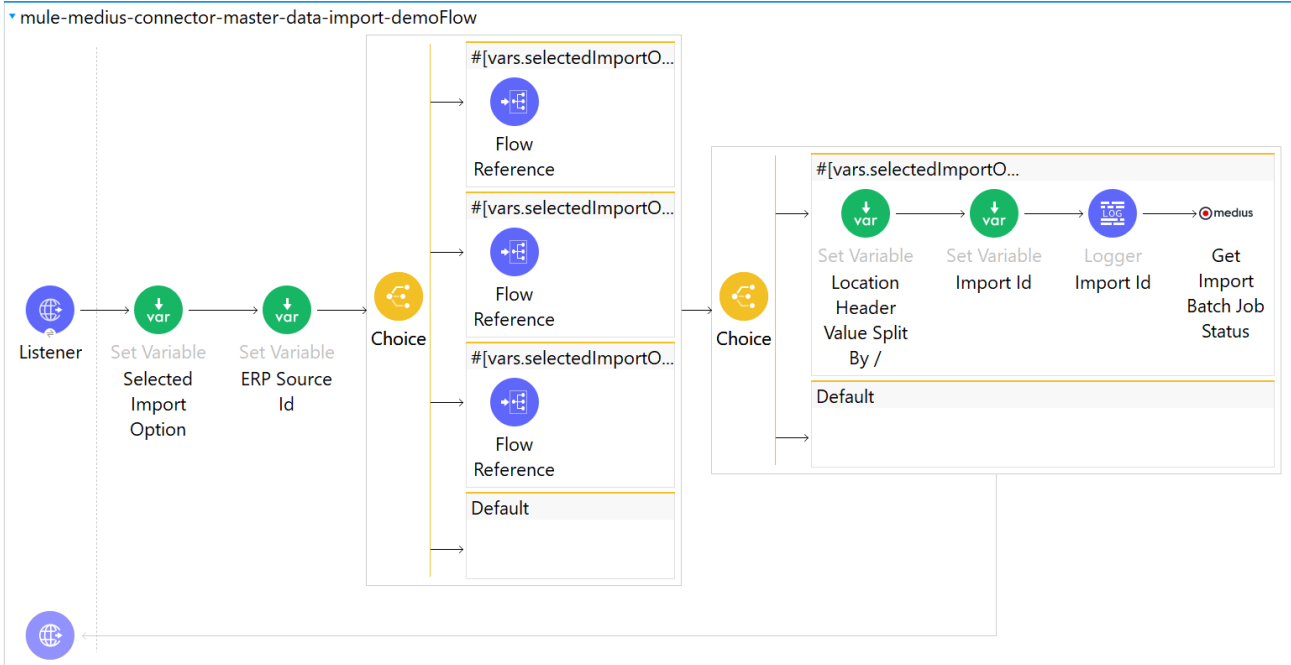
```

        "Description": "Payment",
        "PaymentDetailId": vars.currencyCode,
        "Identifiers": [
            {
                "ErpSourceId": vars.erpSourceId,
                "Value": "123456789",
                "ExternalSystemId": ""
            },
            {
                "ErpSourceId": vars.erpSourceId,
                "Value": vars.currencyCode,
                "ExternalSystemId": ""
            }
        ],
        "IsActive": true,
        "Default": true
    }
]
}
]

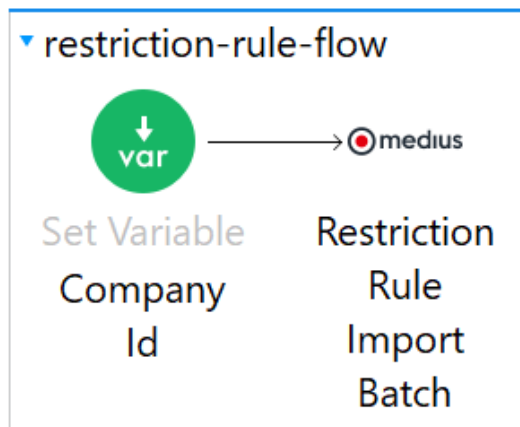
```

- This operation allows the client to send suppliers that will be imported by the Medius master data import queue. The supplier has payment terms and company as mandatory prerequisite. If additional dimensions are going to be used or Vat groups they need to be imported in advance.
- After creating the flow, right-click the project and click **Run As > Mule Application**.
- Once the application is deployed, use **HTTP Listener's Path** to execute the Mule Flow.

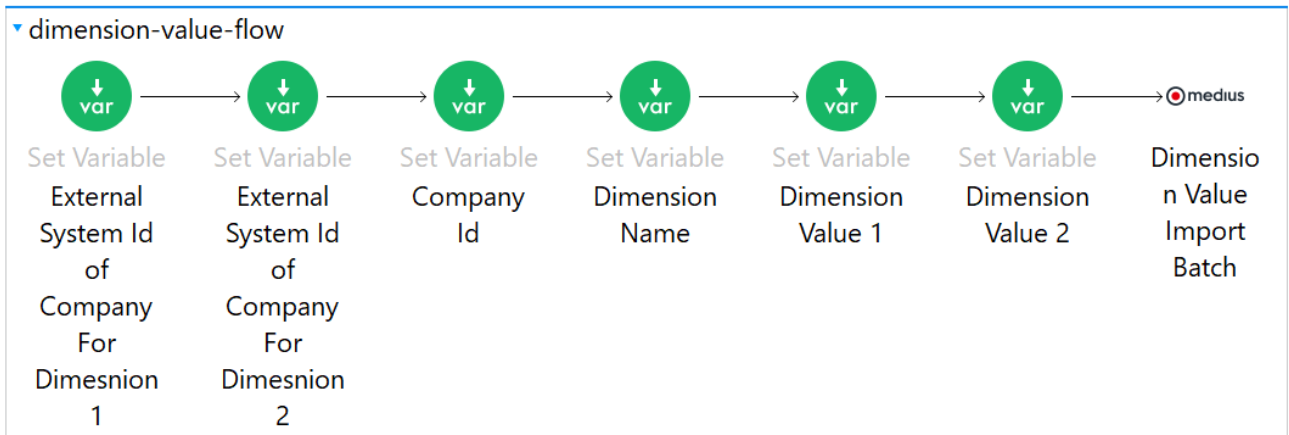
**mule-medijs-connector-master-data-import-demoFlow**



## restriction-rule-flow



## dimension-value-flow



[illegible]

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

<mule
  xmlns:mule4-mediust-connector="http://www.mulesoft.org/schema/mule/mule4-mediust-
connector"
  xmlns:http="http://www.mulesoft.org/schema/mule/http"
  xmlns="http://www.mulesoft.org/schema/mule/core"
  xmlns:doc="http://www.mulesoft.org/schema/mule/documentation"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.mulesoft.org/schema/mule/core
http://www.mulesoft.org/schema/mule/core/current/mule.xsd
http://www.mulesoft.org/schema/mule/http
http://www.mulesoft.org/schema/mule/http/current/mule-http.xsd
http://www.mulesoft.org/schema/mule/mule4-mediust-connector
http://www.mulesoft.org/schema/mule/mule4-mediust-connector/current/mule-mule4-mediust-
connector.xsd">

  <configuration-properties file="mule-application.properties" />

  <http:listener-config name="HTTP_Listener_config"
    doc:name="HTTP Listener config"
    doc:id="09f5cca2-7cd3-437e-a268-5ee0c7824a3d">
    <http:listener-connection host="0.0.0.0"
      port="8081" />
  </http:listener-config>

  <mule4-mediust-connector:config name="Mediust_Connector_Mule4_Config" doc:name="Mediust
Connector Mule4 Config" doc:id="52198e47-aa7a-4d7f-ac92-4489fa2fe8f7" >
    <mule4-mediust-connector:mediustflowcloudauth-connection
      baseUrl="${mediust.base.url}">
      <mule4-mediust-connector:oauth-client-credentials
        clientId="${mediust.client.id}" tokenUrl="${mediust.token.url}" clientSecret="${mediust.client.secret}"
        scopes="${mediust.scopes}" />
    </mule4-mediust-connector:mediustflowcloudauth-connection>
  </mule4-mediust-connector:config>
</mule>
```

```

<flow name="mule-medius-connector-master-data-import-demoFlow"
  doc:id="44b03b34-9f23-49a9-b06c-359d8bc8bac2">
  <http:listener doc:name="Listener"
    doc:id="a25dfd77-e60b-4e52-9d74-97b820122e98"
    config-ref="HTTP_Listener_config" path="/master-data-import">
  </http:listener>
  <set-variable value="{selected.import.option}"
    doc:name="Selected Import Option" doc:id="71499e5f-32a7-44d4-b63a-
02618b4dc779"
    variableName="selectedImportOption" />
  <set-variable value="{erp.source.id}" doc:name="ERP Source Id" doc:id="d86d96b1-
98d6-4ad6-b823-80c9f409893c" variableName="erpSourceId"/>

  <choice doc:name="Choice"
    doc:id="6410c523-7f56-4d96-b54f-d9c13a4a8951">
    <when expression="#[vars.selectedImportOption == 'RESTRICTION_RULE']">
      <flow-ref doc:name="Flow Reference"
        doc:id="fcef4eef-90ad-473a-9465-
4259b3312aa4"
        name="restriction-rule-flow" />

    </when>
    <when expression="#[vars.selectedImportOption == 'DIMENSION_VALUE']">
      <flow-ref doc:name="Flow Reference"
        doc:id="8d4a72ed-600a-4d2f-a8c1-
71aaabdc823f"
        name="dimension-value-flow" />

    </when>
    <when expression="#[vars.selectedImportOption == 'SUPPLIER']">
      <flow-ref doc:name="Flow Reference"
        doc:id="a1b39a34-ad81-4f97-b94c-
3bc6ae6ca8d3"
        name="supplier-flow" />

    </when>
  </choice>

  <choice doc:name="Choice"
    doc:id="f4e07c2a-a9fe-4cee-9cc2-b0ae8377b214">
    <when expression="#[vars.selectedImportOption == 'RESTRICTION_RULE' or
vars.selectedImportOption == 'DIMENSION_VALUE' or vars.selectedImportOption == 'SUPPLIER']">
      <set-variable value="#[attributes.headers.location splitBy ('/')]
doc:name="Location Header Value Split By /" doc:id="ed84773b-cc6a-4ec0-a114-a7c9d80b1a54"
variableName="location" />

      <set-variable value="#[vars.location[-1]]" doc:name="Import Id"
doc:id="26dde20c-bed2-449e-a744-76dd0d0aa935" variableName="importId"/>
      <logger level="INFO" doc:name="Import Id" doc:id="f08ffd5b-3f82-
4f43-a2d0-c51d1f32f018" message="#[vars.importId]"/>
    </when>
  </choice>

```

```

<mule4-medius-connector:get-import-batch-job-status
masterDataTypeUriParam="#[vars.selectedImportOption]" doc:name="Get Import Batch Job Status"
doc:id="e99e50a4-91b2-41e1-b27a-01955470ad34"
config-ref="Medius_Connector_Mule4_Config"
importIdUriParam="#[vars.importId]"/>
</when>
</choice>
</flow>

<sub-flow
name="restriction-rule-flow"
doc:id="be0b9f5d-0fbf-46e0-9412-6a67be779de9">

<set-variable value="{company.id}" doc:name="Company Id" doc:id="3ae2f11e-d3b6-
4f77-8397-e85218d6d666" variableName="companyId" />
<mule4-medius-connector:restriction-rule-import-batch doc:name="Restriction Rule
Import Batch" doc:id="9e60946b-b00f-4753-9a7c-b4b21866fc8d" config-
ref="Medius_Connector_Mule4_Config">
<mule4-medius-connector:restriction-rule-import-batch-bodies
><![CDATA[#[%dw 2.0
output application/json
---
[
{
"ErpSourceId": vars.erpSourceId,
"ExternalSystemId": "ExternalSystemIdBatchNo1",
"CompanyId": vars.companyId,
"IsActive": true,
"RestrictionRules": [
{
"ControllingDimension": "DIMENSION1",
"ControllingDimensionValues": [
{
"Is": "1150"
}
],
"RestrictedOrderedDimensionGroups": [
{
"Dimension": "DIMENSION2",
"ChildRules": [
{
"Type": "forbidden",
"DimensionValues": {
"IsEmptyValue": true
}
}
]
}
]
}
]
},

```

```

{
  "ControllingDimension": "DIMENSION1",
  "ControllingDimensionValues": [
    {
      "Is": "Q103"
    }
  ],
  "RestrictedOrderedDimensionGroups": [
    {
      "Dimension": "DIMENSION3",
      "ChildRules": [
        {
          "Type": "fixed",
          "DimensionValues": {
            "Is": "Q303"
          }
        }
      ]
    }
  ]
},
{
  "ControllingDimension": "DIMENSION2",
  "ControllingDimensionValues": [
    {
      "IsAnyValue": true
    }
  ],
  "RestrictedOrderedDimensionGroups": [
    {
      "Dimension": "DIMENSION3",
      "ChildRules": [
        {
          "Type": "forbidden",
          "DimensionValues": {
            "IsEmptyValue": true
          }
        }
      ]
    }
  ]
},
{
  "ControllingDimension": "DIMENSION1",
  "ControllingDimensionValues": [
    {
      "Is": "1111"
    }
  ],
  "RestrictedOrderedDimensionGroups": [

```

```

{
  "Dimension": "DIMENSION2",
  "ChildRules": [
    {
      "Type": "forbidden",
      "DimensionValues": {
        "IsAnyValue": true
      }
    }
  ]
},
    {
      "Dimension": "DIMENSION3",
      "ChildRules": [
        {
          "Type": "required",
          "DimensionValues": {
            "From": "22",
            "To": "24",
            "IncludeEdges": true
          }
        }
      ]
    }
  ],
  {
    "ControllingDimension": "DIMENSION1",
    "ControllingDimensionValues": [
      {
        "Is": "1112"
      }
    ],
    "RestrictedOrderedDimensionGroups": [
      {
        "Dimension": "DIMENSION2",
        "ChildRules": [
          {
            "Type": "optional",
            "DimensionValues": {
              "IsEmptyValue": true
            }
          }
        ]
      }
    ],
    {
      "Dimension": "DIMENSION3",
      "ChildRules": [
        {
          "Type": "fixed",

```

```

        "DimensionValues": {
          "Is": "22"
        }
      }
    ]
  }
]
},
{
  "ErpSourceId": vars.erpSourceId,
  "ExternalSystemId": "ExternalSystemIdBatchNo2",
  "CompanyId": vars.companyId,
  "IsActive": true,
  "RestrictionRules": [
    {
      "ControllingDimension": "DIMENSION1",
      "ControllingDimensionValues": [
        {
          "Is": "1120"
        }
      ],
      "RestrictedOrderedDimensionGroups": [
        {
          "Dimension": "DIMENSION2",
          "ChildRules": [
            {
              "Type": "forbidden",
              "DimensionValues": {
                "IsEmptyValue": true
              }
            }
          ]
        }
      ]
    }
  ]
}
]
}
]
}
]]]]></mule4-mediuss-connector:restriction-rule-import-batch-bodies>
      </mule4-mediuss-connector:restriction-rule-import-batch>

    </sub-flow>

    <sub-flow
      name="dimension-value-flow"
      doc:id="b8507d56-85f3-40dd-bdc4-c195f4eed502">
      <set-variable
value="$${external.system.company.id};${dimension.name};${dimension.value.1}"
doc:name="External

```

```

System Id of Company For Dimesnion 1" doc:id="d8c74a25-d056-4054-99fd-dac4d418cbac"
variableName="externalSysIdOfCompany1" />
    <set-variable
value="$${external.system.company.id};${dimension.name};${dimension.value.2}" doc:name="External
System Id of Company For Dimesnion 2" doc:id="676610f8-1662-4395-96f5-58b985065c93"
variableName="externalSysIdOfCompany2" />
    <set-variable value="$${company.id.for.dimension.value}" doc:name="Company Id"
doc:id="f12e0cf5-7aa3-4bf0-aa5b-9204583e476f" variableName="company" />
    <set-variable value="$${dimension.name}" doc:name="Dimension Name"
doc:id="ee38571b-b48f-4ebb-b3e3-73fe5d4a33ce" variableName="dimensionName" />
    <set-variable value="$${dimension.value.1}" doc:name="Dimension Value 1"
doc:id="f4457f2d-93ff-4595-9566-510db8294605" variableName="dimensionVal1" />
    <set-variable value="$${dimension.value.2}" doc:name="Dimension Value 2"
doc:id="874a98c1-8e16-4fbc-b18b-230c34a9a7ff" variableName="dimensionVal2" />

    <mule4-mediuis-connector:dimension-value-import-batch doc:name="Dimension Value
Import Batch" doc:id="3f5b13a5-cb4f-4920-996d-d607698328ec" config-
ref="Medius_Connector_Mule4_Config">
        <mule4-mediuis-connector:dimension-value-import-batch-bodies
><![CDATA[#[%dw 2.0
output application/json
---
[
    {
        "ErpSourceId": vars.erpSourceId,
        "ExternalSystemId": vars.externalSysIdOfCompany1,
        "IsActive": true,
        "CompanyId": vars.company,
        "DimensionName": vars.dimensionName,
        "Description": "Internal purchases",
        "Value": vars.dimensionVal1,
        "TaxIndicator1": "",
        "TaxIndicator2": "",
        "IsFictive": true,
        "UsableInPurchase": true
    },
    {
        "ErpSourceId": vars.erpSourceId,
        "ExternalSystemId": vars.externalSysIdOfCompany2,
        "IsActive": true,
        "CompanyId": vars.company,
        "DimensionName": vars.dimensionName,
        "Description": "External purchases",
        "Value": vars.dimensionVal2,
        "TaxIndicator1": "",
        "TaxIndicator2": "",
        "IsFictive": true,
        "UsableInPurchase": true
    }
}]]]></mule4-mediuis-connector:dimension-value-import-batch-bodies>

```

```

        </mule4-medijs-connector:dimension-value-import-batch>
    </sub-flow>

    <sub-flow
        name="supplier-flow"
        doc:id="a8f255b1-f74d-415f-bccb-963880270185">

        <set-variable value="\${external.system.company.id};\${supplier.id}" doc:name="External
        System Id of Company" doc:id="f2384150-d473-427b-bb10-e03fe81a1ea6" variableName="externalSysId" />
        <set-variable value="\${company.id.for.dimension.value}" doc:name="Company Id"
        doc:id="67632039-8dbc-4f19-8c6d-b8f29a20850e" variableName="company" />
        <set-variable value="\${supplier.id}" doc:name="Supplier Id" doc:id="6f22ee1a-f0ec-
        4aa1-aaa8-442586807f71" variableName="supplierId" />
        <set-variable value="\${currency.code}" doc:name="Currency Code" doc:id="494100e5-
        e326-4a06-840a-524a7e30f77b" variableName="currencyCode" />
        <set-variable value="\${delivery.term}" doc:name="Delivery Term" doc:id="55e183e5-
        4467-4ea1-a09f-832757ba80f2" variableName="deliveryTerm" />
        <set-variable value="\${external.system.company.id};\${payment.term.id}"
        doc:name="Payment Term Id" doc:id="2bd6bfa3-2e87-41ba-b74e-a394e1af4792"
        variableName="paymentTermId" />
        <set-variable
        value="\${external.system.company.id};\${ledger.dimension};\${ledger.dimension.value}" doc:name="Ledger
        Dimension" doc:id="e3ffb252-7748-41b4-a5d1-9c0dd4fc2f39" variableName="ledgerDimension" />
        <set-variable
        value="\${external.system.company.id};\${tax.dimension};\${tax.dimension.value}" doc:name="Tax
        Dimension" doc:id="6e6f4a74-d45c-423f-abe1-8317540282de" variableName="taxDimension" />
        <set-variable
        value="\${external.system.company.id};\${clearence.dimension};\${clearence.dimension.value}"
        doc:name="Clearence Dimension" doc:id="3a52e2a8-a6f4-408c-9621-e0fb01407d4f"
        variableName="clearenceDimension" />
        <set-variable value="\${external.system.company.id};\${tax.group.code}" doc:name="Tax
        Group Code" doc:id="ee32759f-9936-4533-b2b2-1355ee8c4646" variableName="taxGroupCode" />
        <set-variable value="\${external.system.company.id};\${supplier.id};\${currency.code}"
        doc:name="External System Id of Company" doc:id="ddfafc25-3920-4ac2-9840-4ea7d4d0d124"
        variableName="externalSysIdPaymentDetail" />

        <mule4-medijs-connector:supplier-import-batch doc:name="Supplier Import Batch"
        doc:id="0726a255-c1f1-41cf-91fc-217caea3ea33" config-ref="Medius_Connector_Mule4_Config">
            <mule4-medijs-connector:supplier-import-batch-bodies ><![CDATA[#[%dw
2.0
output application/json
---
[
    {
        "ErpSourceId": vars.erpSourceId,
        "ExternalSystemId": vars.externalSysId,
        "IsActive": true,
        "CompanyId": vars.company,
        "CurrencyCode": vars.currencyCode,
        "Identifiers": [

```

```

{
  "IdentifierType": "FINANCIAL_SYSTEM",
  "IdentifierValue": vars.supplierId
},
{
  "IdentifierType": "BANK_ACCOUNT",
  "IdentifierValue": "123456789"
},
{
  "IdentifierType": "ORGANIZATION_NUMBER",
  "IdentifierValue": "99999999"
}
],
"Name": "TestSupplier5",
"PaymentTerm": vars.paymentTermId,
"DeliveryTerm": vars.deliveryTerm,
"SupplierId": vars.supplierId,
"LedgerDimension1": vars.ledgerDimension,
"TaxDimension1": vars.taxDimension,
"Clearance1Dimension1": vars.clearanceDimension,
"TaxIndicator1": vars.taxGroupCode,
"TaxIndicator2": "",
"StreetAddress": "TestStreet",
"City": "TestCity",
"Zip": "0000",
"Country": "TestCountry",
"Telephone": "123456",
"Fax": "",
"Homepage": "",
"EmailAddress": "",
"State": "",
"IsProcurable": true,
"Prepayment": false,
"PaymentDetails": [
{
  "ExternalSystemId": vars.externalSysIdPaymentDetail,
  "Description": "Payment",
  "PaymentDetailId": vars.currencyCode,
  "Identifiers": [
    {
      "ErpSourceId": vars.erpSourceId,
      "Value": "123456789",
      "ExternalSystemId": ""
    },
    {
      "ErpSourceId": vars.erpSourceId,
      "Value": vars.currencyCode,
      "ExternalSystemId": ""
    }
  ]
}
],

```

```

        "IsActive": true,
        "Default": true
      }
    ]
  }
]]]]></mule4-mediust-conector:supplier-import-batch-bodies>
      </mule4-mediust-conector:supplier-import-batch>
    </sub-flow>

</mule>

```

## Send Payment Batch and Fetch Payment Batch Reconciliation Status and Payment Items

This use case demonstrates the usage of various payment batch operations based on selection of flow to execute.

- In Anypoint Studio, click **File > New > Mule Project**, name the project **mule-mediust-conector-payment-demo** and click Finish.
- In the search field, type **Http** and drag the **HTTP Listener** to the canvas.
- Click the **HTTP Listener**, click the green plus sign to the right of Connector Configuration. On the next screen, add the host and port, click OK
- Provide **Path** to HTTP Listener (Example: **/payment-batch**)
- On the Mule Palette, search for **Set Variable** and drag it onto the canvas. Set value of variable from **mule-application.properties** file key “**selected.option**” as **\${selected.option}** and give name of variable as **selectedOpt** and display name as **Selected Option**.
- There are three options available for execution of three different flows. This option user can change from **mule-application.properties** file.
  - **GET\_PAYMENT\_BATCH\_STATUS** - Get Payment Batch Reconciliation Status By Payment Batch Id
  - **GET\_PAYMENT\_BATCH\_STATUS\_BY\_EXTERNAL\_BATCH\_ID** - Get Payment Batch Reconciliation Status By External Batch Id
  - **POST\_PAYMENT\_BATCH** - Send Payment Batches
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- There will be three **When** blocks for added **Choice**. Those will be as follows.
- Add below condition as expression in **Choice** to check if user wants to execute get payment batch reconciliation status by payment batch id operations related flow.

```
vars.selectedOpt == 'GET_PAYMENT_BATCH_STATUS'
```

- Add **Flow Reference** to call another sub flow '**payment-batch-status-flow**'.
- Add below condition as expression in **Choice** to check if the user wants to execute get payment batch reconciliation status by external batch id operations related flow.

```
vars.selectedOpt == 'GET_PAYMENT_BATCH_STATUS_BY_EXTERNAL_BATCH_ID'
```

- Add **Flow Reference** to call another sub flow '**payment-batch-status-by-external-batch-id-flow**' to execute Send Record to ERP and then External System Procurement Goods Receipt Note Confirmation operation related flow.
- Add below condition as expression in **Choice** to check if the user wants to send a payment batch.

```
vars.selectedOpt == 'POST_PAYMENT_BATCH'
```

- Add **Flow Reference** to call another sub flow '**post-payment-batch-flow**'.
- Create a new sub flow **payment-batch-status-flow**.
- Add **Set Variable** and set variable name as **paymentBatchId** and value as **\${payment.batch.id}** key from **mule-application.properties** file and display name as **Payment Batch Id**.
- Add **Get Payment Batch Reconciliation Status By Payment Batch Id** Medius operation from **Mule Palette** to canvas and provide **payment batch id** as expression **vars.paymentBatchId**. This operation gets payment batch reconciliation status using payment batch id. Basic information about payment batches.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if status code is 200.

```
attributes.statusCode == 200
```

- Add **Get Payment Items Reconciliation Status** Medius operation from **Mule Palette** to canvas and provide **payment batch id** as expression **vars.paymentBatchId**. This operation gets payment items reconciliation status. Detailed information about the items of the payment batch.
- Create a new sub flow **payment-batch-status-by-external-batch-id-flow**.
- Add **Set Variable** and set variable name as **externalBatchId** and value as **\${external.payment.batch.id}** key from **mule-application.properties** file and display name as **External Batch Id**.
- Add **Get Payment Batch Reconciliation Status By External Batch Id** Medius operation from **Mule Palette** to canvas and provide **external batch id** as expression **vars.externalBatchId**. This

operation gets payment batch reconciliation status using external batch id. Basic information about payment batches.

- Create a new sub flow **post-payment-batch-flow**.
- Add **Try** block named **Try - Block for Send Payment Batch**.
- In the try block, Add **Set Variable** and set variable name as **externalBatchId** and value as **\${external.payment.batch.id}** key from **mule-application.properties** file and display name as **External Batch Id**.
- Add **Set Variable** and set variable name as **payorId** and value as **\${payor.id}** key from **mule-application.properties** file and display name as **Payor Id**.
- Add **Send Payment Batches** Medius operation from **Mule Palette** to canvas and provide **body** as expression : **(Here you can provide values as per your data)**

```
%dw 2.0
output application/json
---
{
  "payorId": vars.payorId,
  "externalBatchId": vars.externalBatchId,
  "items": [
    {
      "invoiceNumber": "2/2021",
      "voucherNumber": "1519c142-c405-4753-bb2f-5fbec59375d4",
      "invoiceCompanyId": "Medius",
      "payeeId": "sinid",
      "remitToID": "345123R",
      "payeeName": "Larson - Roob",
      "payeeStreetLine1": "3346 Pagac Union",
      "payeeStreetLine2": "PO box 1202",
      "payeeCity": "Lake Miller",
      "payeeZipOrPostalCode": "89262-8263",
      "payeeStateCode": "RI",
      "payeeCountryCode": "US",
      "paymentAmount": 732.43,
      "paymentCurrencyCode": "USD",
      "invoiceDate": "2021-04-10",
      "invoiceDueDate": "2021-04-10",
      "invoiceAmount": 732.43,
```

```

    "invoiceCurrencyCode": "USD",
    "payorBankAccountIdentifier": "BankId1",
    "payorLocationIdentifier": "1",
    "comment": "Comment for payee",
    "payorIdInPayeeSystem": "897578762",
    "settlementCurrencyCode": "USD",
    "payeeBankName": "Bank Zachodni WBK S.A.",
    "payeeBankStreetLine1": "PL.WOLNOSCI 15",
    "payeeBankStreetLine2": "BUILDING B",
    "payeeBankCity": "POZNAN",
    "payeeBankCountryCode": "PL",
    "payeeAccountNumber": "PL88109013620000000036000000",
    "payeeRoutingNumber": "202020",
    "payeeBankSwiftCode": "WBKPPLPP",
    "purposeOfPayment": "BillPayment",
    "payeeType": "Business",
    "payeeContactPhone": "123456789",
    "payeeContactEmail": "email@email.com",
    "paymentReference": "P012701",
    "discountAmount": 0,
    "discountCurrencyCode": "USD"
  }
]
}

```

- This operation allows the client to send payment batches that will be imported into Medius Pay to be approved and paid. When import fails, an incorrect payment batch is saved in the Import error inbox within Medius Pay.
- In the Error Handling block, add **On Error Propagate** from **Mule Palette** and provide type as **ANY**
- In **On Error Propagate** block, add **Logger** to log error and set display name as **Error** and message as expression **error**
- After **Try** block, Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check the status code is 201.

```
attributes.statusCode == 201
```

- Add **Set Variable** and set display name as **Payment Batch Id**, variable name as **paymentBatchId** and

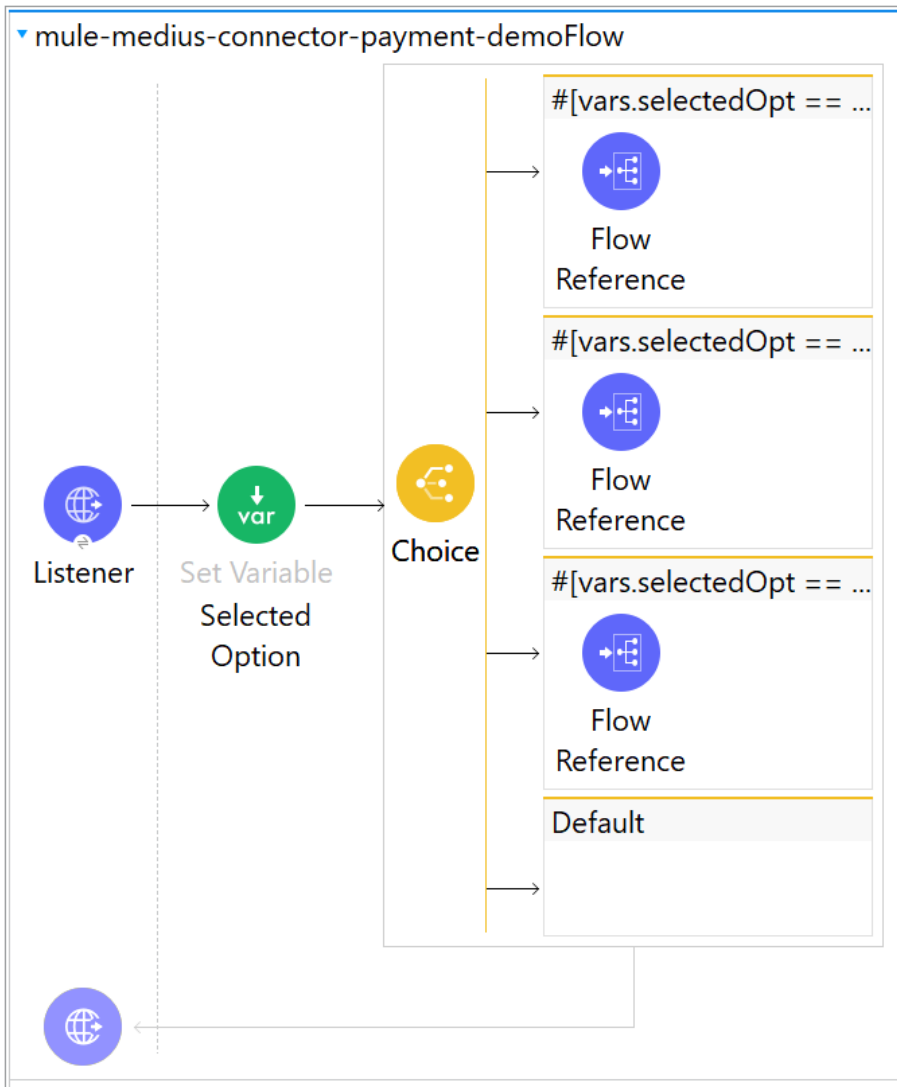
value as expression **payload.paymentBatchId**.

- Add **Get Payment Batch Reconciliation Status By Payment Batch Id** Medius operation from **Mule Palette** to canvas and provide **payment batch id** as expression **vars.paymentBatchId**. This operation gets payment batch reconciliation status using payment batch id. Basic information about payment batches.
- Add **Choice** from Mule Palette to the canvas. It is used to switch between the different choices depending on the expression provided.
- Add below condition as expression in **Choice** to check if status code is 200.

|                                           |
|-------------------------------------------|
| <code>attributes.statusCode == 200</code> |
|-------------------------------------------|

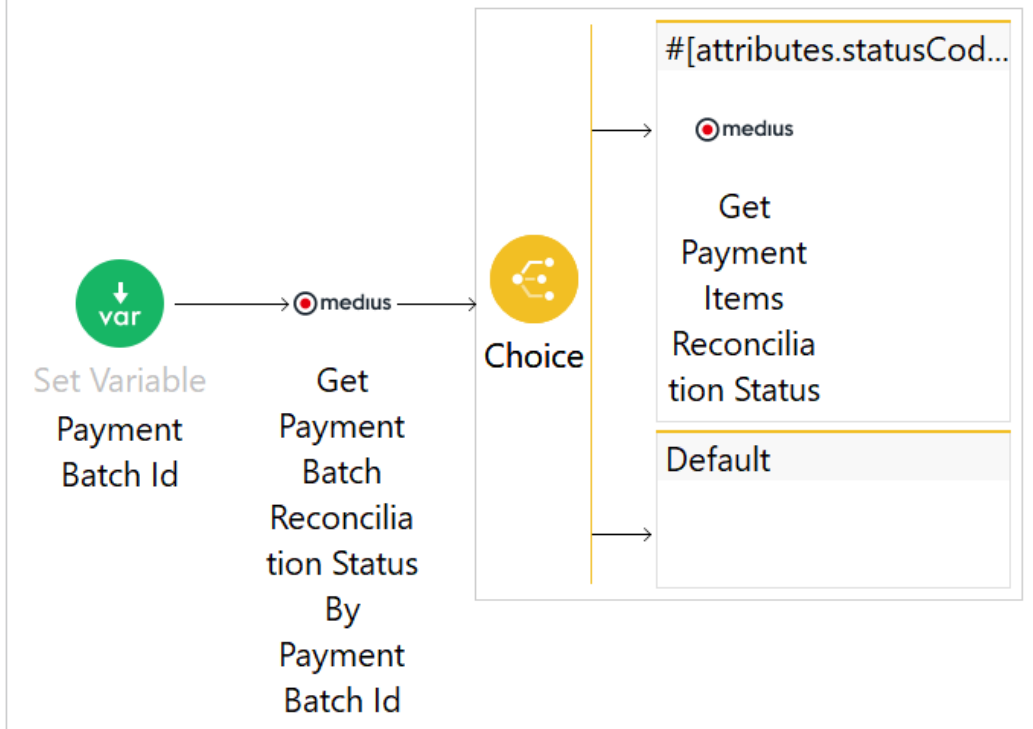
- Add **Get Payment Items Reconciliation Status** Medius operation from **Mule Palette** to canvas and provide **payment batch id** as expression **vars.paymentBatchId**. This operation gets payment items reconciliation status. Detailed information about the items of the payment batch.
- After creating the flow, right-click the project and click **Run As > Mule Application**.
- Once the application is deployed, use **HTTP Listener's Path** to execute the Mule Flow.

**mule-medijs-connector-payment-demoFlow**

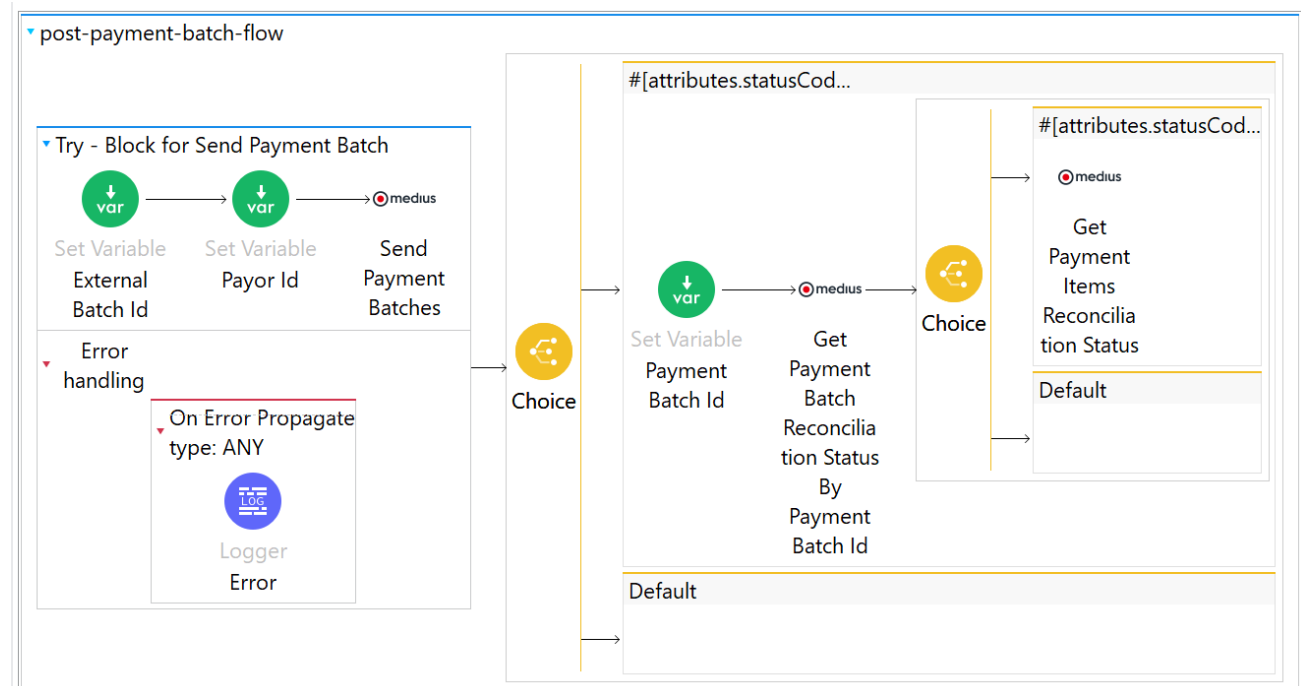


**payment-batch-status-flow**

### ▼ payment-batch-status-flow

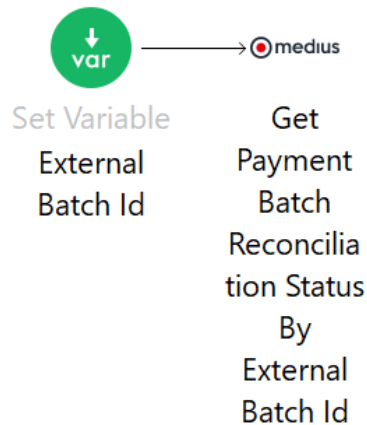


### post-payment-batch-flow



### payment-batch-status-by-external-batch-id-flow

▼ payment-batch-status-by-external-batch-id-flow



**Use Case Configuration XML :** Paste this XML code into Anypoint Studio to experiment with the flow described above.

```

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

<mule
  xmlns:mule4-medijs-connector="http://www.mulesoft.org/schema/mule/mule4-medijs-connector"
  xmlns:http="http://www.mulesoft.org/schema/mule/http"
  xmlns="http://www.mulesoft.org/schema/mule/core"
  xmlns:doc="http://www.mulesoft.org/schema/mule/documentation"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.mulesoft.org/schema/mule/core
http://www.mulesoft.org/schema/mule/core/current/mule.xsd
http://www.mulesoft.org/schema/mule/http
http://www.mulesoft.org/schema/mule/http/current/mule-http.xsd
http://www.mulesoft.org/schema/mule/mule4-medijs-connector/current/mule-mule4-medijs-connector.xsd">

  <configuration-properties
    file="mule-application.properties" />

  <http:listener-config name="HTTP_Listener_config"
    doc:name="HTTP Listener config"
    doc:id="a2612d16-1270-488c-a6b4-6954c24d1762">
    <http:listener-connection host="0.0.0.0"
      port="8081" />
  </http:listener-config>

  <mule4-medijs-connector:config
    name="Medius_Connector_Mule4_Config"
  
```

```

doc:name="Medius Connector Mule4 Config"
doc:id="f6440a0b-37db-4a30-952f-c1d306c7fd22">
<mule4-medijs-connector:medijsflowcloudauth-connection
    baseUri="{medijs.base.url}">
    <mule4-medijs-connector:oauth-client-credentials
        clientId="{medijs.client.id}" tokenUrl="{medijs.token.url}"
        clientSecret="{medijs.client.secret}" scopes="{medijs.scopes}"
    />
/>

</mule4-medijs-connector:medijsflowcloudauth-connection>
</mule4-medijs-connector:config>

<flow name="mule-medijs-connector-payment-demoFlow"
    doc:id="d1867059-04f5-4108-ba8c-c664799e705f">

    <http:listener doc:name="Listener"
        doc:id="8885df8b-423b-4a52-8a31-324c46860f8d"
        config-ref="HTTP_Listener_config" path="/payment-batch" />
    <set-variable value="{selected.option}"
        doc:name="Selected Option"
        doc:id="cf6b4f2e-4ec7-4610-9e5b-ccfb1ceba525"
        variableName="selectedOpt" />

    <choice doc:name="Choice"
        doc:id="3695ee03-17f1-4861-96c6-344499d334d1">
        <when
            expression="#[vars.selectedOpt ==
'GET_PAYMENT_BATCH_STATUS']">

            <flow-ref doc:name="Flow Reference"
                doc:id="9c91580b-91db-4bfb-afb5-b48b819ab45a"
                name="payment-batch-status-flow" />

        </when>
        <when
            expression="#[vars.selectedOpt ==
'GET_PAYMENT_BATCH_STATUS_BY_EXTERNAL_BATCH_ID']">

            <flow-ref doc:name="Flow Reference"
                doc:id="3c1c6e4a-d023-4c6c-b2f2-5919c5cc8504"
                name="payment-batch-status-by-external-batch-id-flow"
            />

        </when>
        <when expression="#[vars.selectedOpt == 'POST_PAYMENT_BATCH']">

            <flow-ref doc:name="Flow Reference"
                doc:id="0efdd234-b2ee-49da-90df-6180bcc2c978"
                name="post-payment-batch-flow" />

        </when>

    </choice>

```

```

</flow>

<sub-flow name="payment-batch-status-flow"
  doc:id="e19ae7e0-ae45-4bee-b2ec-5f5299dc2bfa">
  <set-variable value="{payment.batch.id}"
    doc:name="Payment Batch Id"
    doc:id="61097192-fb64-4eb3-98cf-6446a29b7b36"
    variableName="paymentBatchId" />
  <mule4-medijs-connector:get-payment-batch-reconciliation-status-by-payment-batch-
id
    paymentBatchIdUriParam="#[vars.paymentBatchId]"
    doc:name="Get Payment Batch Reconciliation Status By Payment Batch Id"
    doc:id="8b889293-3570-4428-83de-fb8a8ac5d874"
    config-ref="Medius_Connector_Mule4_Config" />
  <choice doc:name="Choice"
    doc:id="5ca6ccc8-d4d8-4a3c-9c44-c748245396dd">
    <when expression="#[attributes.statusCode == 200]">
      <mule4-medijs-connector:get-payment-items-reconciliation-status
        paymentBatchIdUriParam="#[vars.paymentBatchId]"
        doc:name="Get Payment Items Reconciliation Status"
        doc:id="4c7a10eb-8821-45ab-8282-6a0f02ccffea"
        config-ref="Medius_Connector_Mule4_Config" />
    </when>
  </choice>
</sub-flow>

<sub-flow name="post-payment-batch-flow">
  <try doc:name="Try - Block for Send Payment Batch" doc:id="d404df70-8566-4fb9-9ec0-
e76d22bf4c5d">
    <set-variable value="{external.payment.batch.id}" doc:name="External
Batch Id" doc:id="f45102a0-556c-4302-b0c6-9814a53b4a38" variableName="externalBatchId" />
    <set-variable value="{payor.id}" doc:name="Payor Id" doc:id="c95d861c-
efae-41df-b1b5-a38e3c2387ed" variableName="payorId"/>
    <mule4-medijs-connector:send-payment-batches
      doc:name="Send Payment Batches"
      doc:id="af4ad56e-4cec-4e11-8373-31c76a5927b0"
      config-ref="Medius_Connector_Mule4_Config">
      <mule4-medijs-connector:send-payment-batches-
body><![CDATA[#[%dw 2.0
output application/json
---
{
  "payorId": vars.payorId,
  "externalBatchId": vars.externalBatchId,
  "items": [
    {
      "invoiceNumber": "2/2021",
      "voucherNumber": "1519c142-c405-4753-bb2f-5fbec59375d4",
      "invoiceCompanyId": "Medius",
      "payeeId": "sinid",

```

```

"remitToID": "345123R",
"payeeName": "Larson - Roob",
"payeeStreetLine1": "3346 Pagac Union",
"payeeStreetLine2": "PO box 1202",
"payeeCity": "Lake Miller",
"payeeZipOrPostalCode": "89262-8263",
"payeeStateCode": "RI",
"payeeCountryCode": "US",
"paymentAmount": 732.43,
"paymentCurrencyCode": "USD",
"invoiceDate": "2021-04-10",
"invoiceDueDate": "2021-04-10",
"invoiceAmount": 732.43,
"invoiceCurrencyCode": "USD",
"payorBankAccountIdentifier": "BankId1",
"payorLocationIdentifier": "1",
"comment": "Comment for payee",
"payorIdInPayeeSystem": "897578762",
"settlementCurrencyCode": "USD",
"payeeBankName": "Bank Zachodni WBK S.A.",
"payeeBankStreetLine1": "PL.WOLNOSCI 15",
"payeeBankStreetLine2": "BUILDING B",
"payeeBankCity": "POZNAN",
"payeeBankCountryCode": "PL",
"payeeAccountNumber": "PL88109013620000000036000000",
"payeeRoutingNumber": "202020",
"payeeBankSwiftCode": "WBKPPLPP",
"purposeOfPayment": "BillPayment",
"payeeType": "Business",
"payeeContactPhone": "123456789",
"payeeContactEmail": "email@email.com",
"paymentReference": "P012701",
"discountAmount": 0,
"discountCurrencyCode": "USD"
}
]
}}]]></mule4-medijs-connector:send-payment-batches-body>
</mule4-medijs-connector:send-payment-batches>
<error-handler>
  <on-error-propagate enableNotifications="true"
    logException="true" doc:name="On Error Propagate"
    doc:id="f616dd93-93f5-4d37-9954-fe6be8b4b836"
    type="ANY">
    <logger level="INFO" doc:name="Error"
      doc:id="ecd3a8fd-0f4b-4198-ba90-
6b2874113c67" message="#[error]" />
  </on-error-propagate>
</error-handler>
</try>
<choice doc:name="Choice"

```

```

doc:id="faf55e5c-20cd-4379-bb10-014dbdf8f0e3">
<when expression="#[attributes.statusCode == 201]">
  <set-variable value="#[payload.paymentBatchId]"
    doc:name="Payment Batch Id"
    doc:id="3d542b99-6f27-454e-aeca-1526576ab39d"
    variableName="paymentBatchId" />
  <mule4-medius-connector:get-payment-batch-reconciliation-
status-by-payment-batch-id
    doc:id="0e2cdd4e-2fef-4c92-b613-53c995cc67a9"
    config-ref="Medius_Connector_Mule4_Config" />
  <choice doc:name="Choice"
    doc:id="e78e3788-05f7-4e90-9e36-9e6cef96e275">
    <when expression="#[attributes.statusCode == 200]">
      <mule4-medius-connector:get-payment-items-
reconciliation-status
    doc:id="fd70be50-2fb8-4904-8bf6-9349392ce596"
    config-ref="Medius_Connector_Mule4_Config" />
    </when>
  </choice>
</when>
</sub-flow>
<sub-flow name="payment-batch-status-by-external-batch-id-flow">
  <set-variable value="$external.payment.batch.id" doc:name="External Batch Id"
doc:id="bb50f3a9-0fa8-41b1-b8d4-812fbd461d16" variableName="externalBatchId"/>
  <mule4-medius-connector:get-payment-batch-reconciliation-status-by-external-batch-
id doc:name="Get Payment Batch Reconciliation Status By External Batch Id" doc:id="cbffd6fc-dcfa-4c36-
b641-e7afcdd7831c"
    config-ref="Medius_Connector_Mule4_Config"
    externalBatchIdQueryParam="#[vars.externalBatchId]"/>
</sub-flow>
</mule>

```

## Resources

- [Medius Connector Release Note](#)
- [Medius Connector API Documentation](#)