|  |
| --- |
| **XML to CSV Convertor in SAP CPI using Multicast join, gather, general splitter and process call.** |
| **(Training Material) Version: 1.0** |

**Table of Contents**

[**Objective:** 2](#_Toc1403306)

[**Challenge:** 2](#_Toc1403307)

[**Solution:** 2](#_Toc1403308)

[**Testing:** 13](#_Toc1403309)

**Document Control**

**Revision History:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Author(s)** | **Reviewer** | **Revision Notes** |
| 1.0 | 18-Feb-2019 | Sidharth VR | Binod Kumar Mondal | First release of document |
|  |  |  |  |  |
|  |  |  |  |  |

# **Objective:**

The Objective of this Interface is to convert a Struct XML message to a CSV File. We will also use Multicast, join, gather, general splitter and process call pallets and understand its uses.

# **Challenge:**

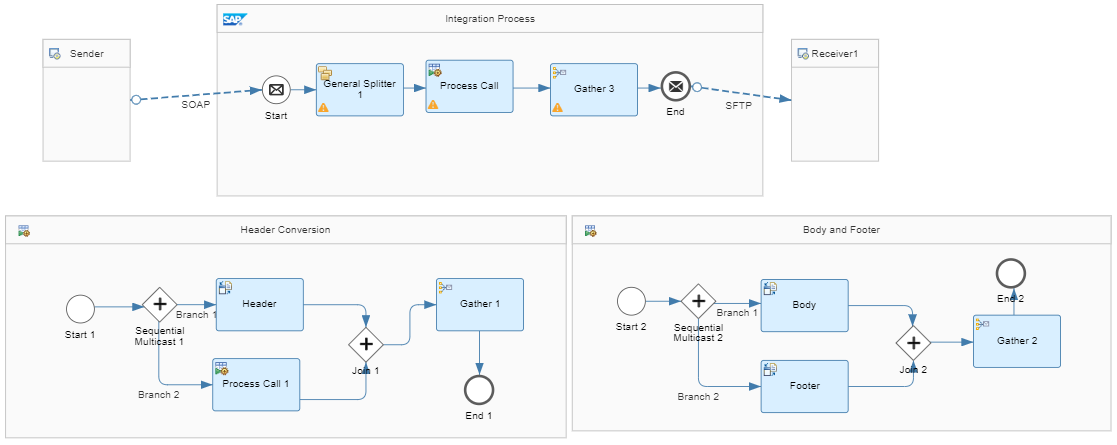
XML to CSV Convertor in SAP CPI will allow to convert only Simple XML to CSV. As an option, we can have the Parent node and child node to get converted. But, if we have Header, Body and Footer, then conversion is not possible directly.

# **Solution:**

This can be handled in CPI in two ways:

1. Writing an XSLT Mapping that will convert an XML to CSV.
2. Constructing Integration flow that can break down the complex structure, convert and then combine the values. (we will use this approach in this demo)

In this flow, we will be demonstrating on how to construct the Iflow to convert Struct XML to CSV format.



Step 1:

Create a package where you can construct the Iflow

Step 2:

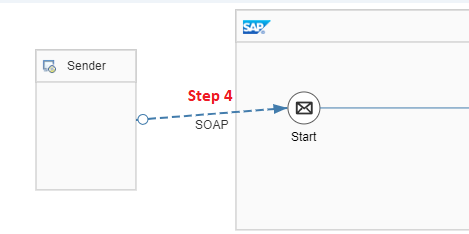
Add Iflow and give a name to the flow using proper naming convention standards.

Step 3:

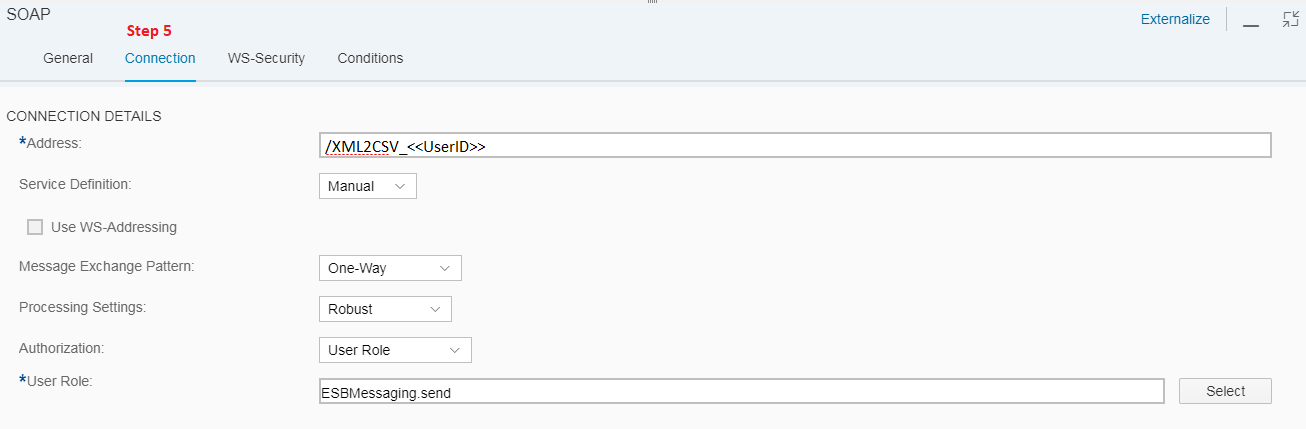
Add a Sender Participant and a Receiver Participant from the Pallet List.

Step 4:

Connect the Sender participant and Start message using SOAP Adapter. (For Test Purpose)

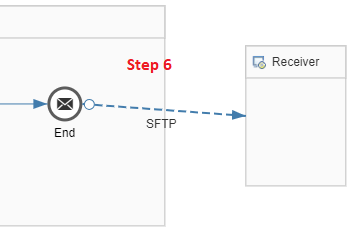


Step 5: Configure the SOAP Adapter in Connection tab as shown below. No changes required for WS-Security and Conditions tab.

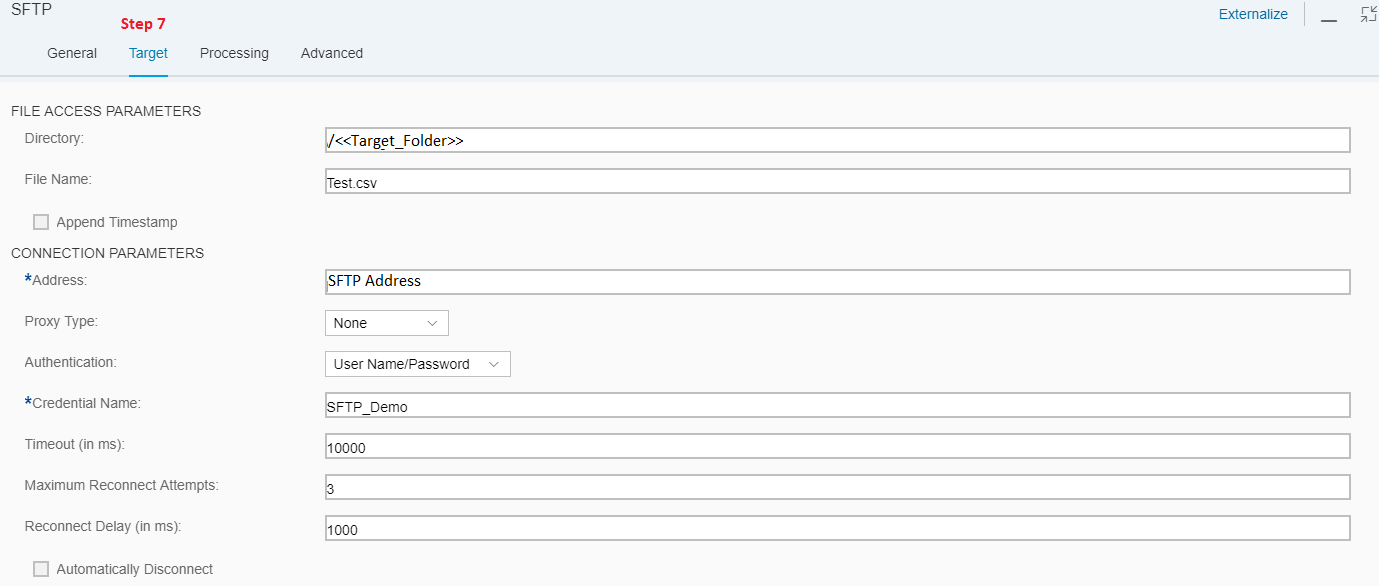


Step 6:

Similarly Connect the End Message and Receiver SFTP Adapter. (Use your own SFTP Server Details)



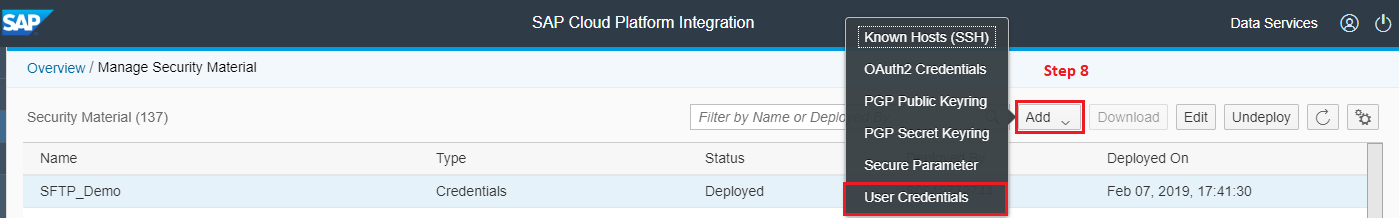
Step 7: Configure the SFTP adapter in Target tab as shown below. No changes required for processing and advanced tab.



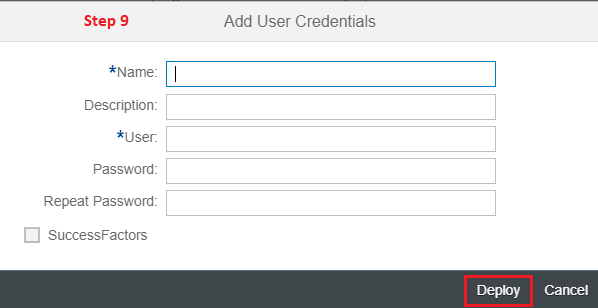
Step 8: SFTP on Premise Credentials should be deployed in the Manage Security Material Tab.

Overview----> Manage Security Material

Click on Add ---> User Credentials

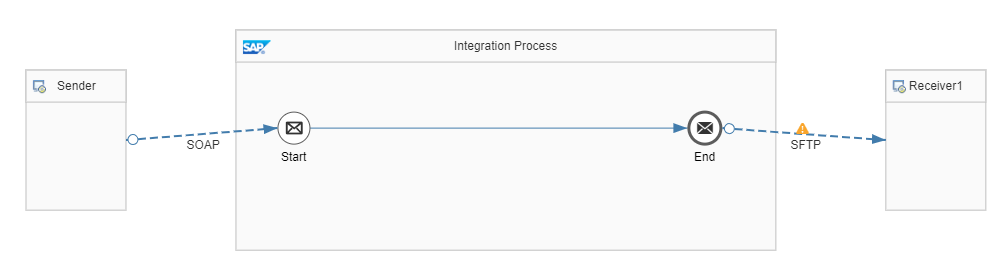


Step 9: Enter the SFTP User ID and password and give the Name as SFTP\_Demo\_<<User ID>>



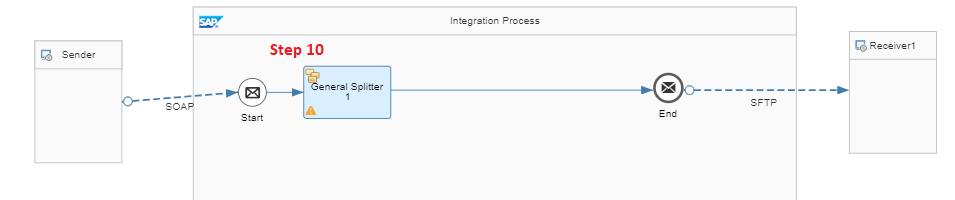
Now this Credential Name will be called in Channel.

Now the flow looks like the below screenshot.



Step 10:

Between Start message and End message place the General Splitter.



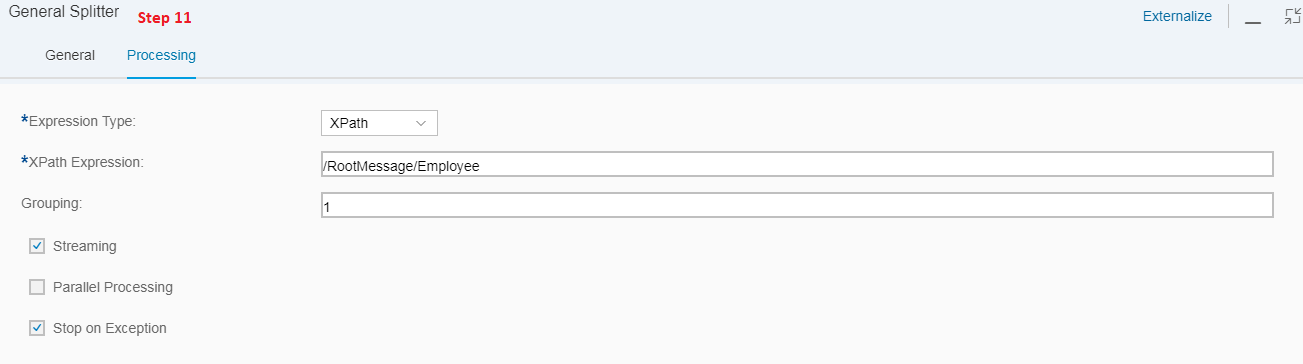
***General Splitter is used to split the multiple repetition of the message into individual entities***.

Step 11: Configure the General Splitter in the Processing tab as show in below screenshot

Xpath must have the root message which needs to be repeated

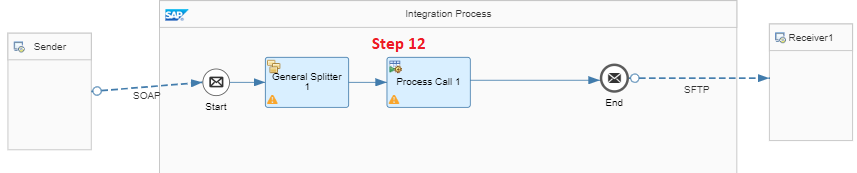
Xpath: /RootMessage/Employee

Grouping: 1

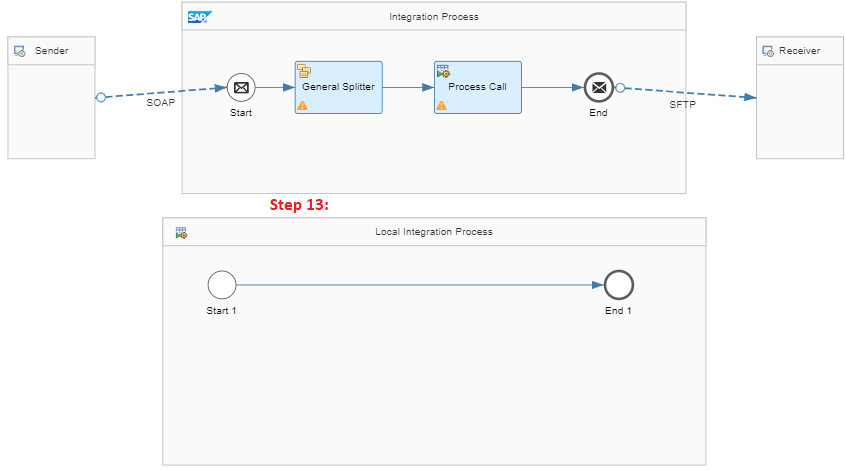
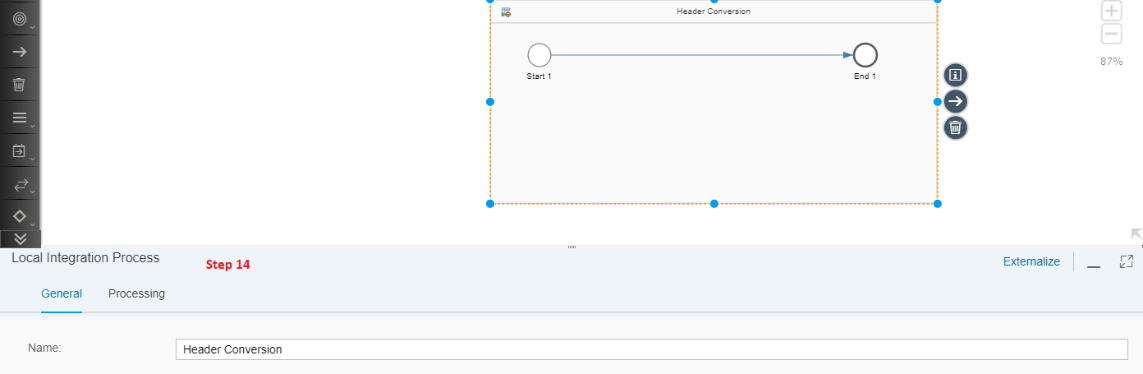


Step 12: From the Palate Choose, Call ---> Local Call ---> Process Call

This needs to be placed between the General Splitter and End Message

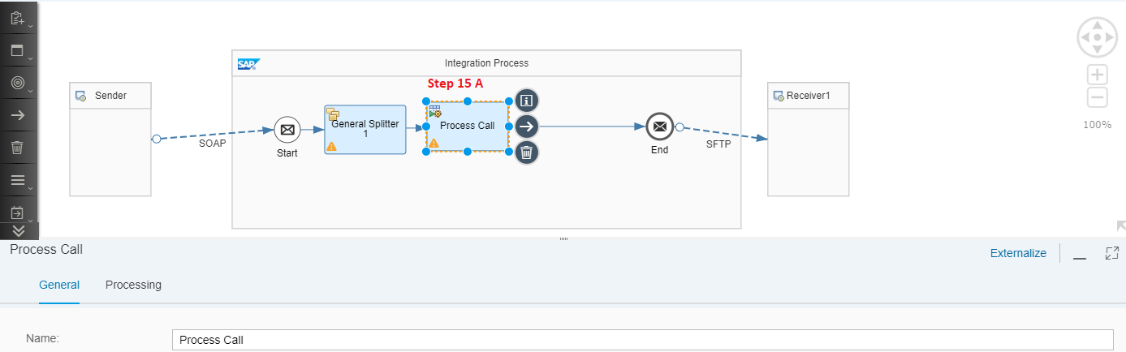


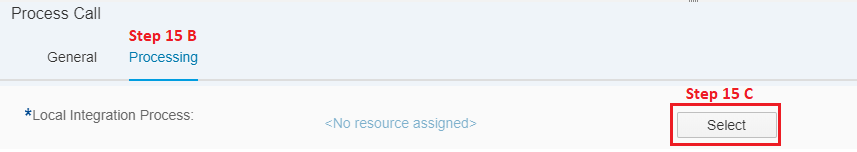
Step 13: Now create a local Integration process from pallet tab. Process-----> Local Integration Process



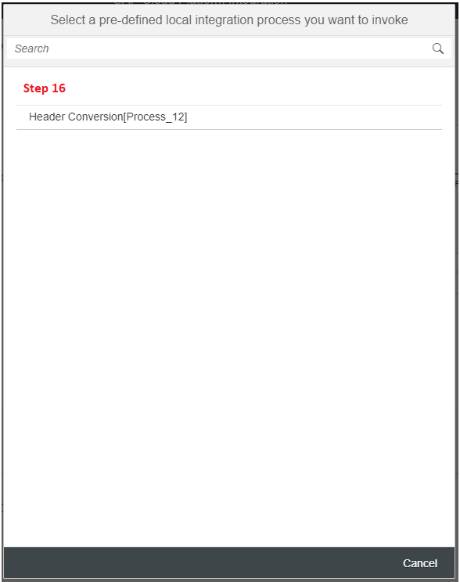
Step 14: Rename the Local Integration Process to Header Conversion

Step 15: Click on Process Call in Integration Process (15 A). Then click on Processing tab (15 B) and click on Select (15 C)

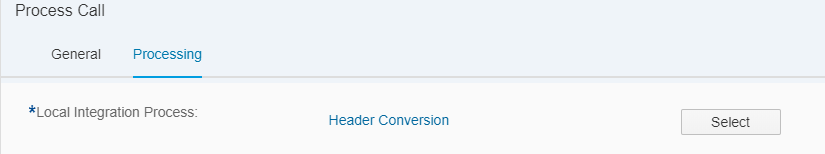




Step 16: A window pops up *“Select a pre-defined local integration process you want to invoke”.* Click on Header Conversion

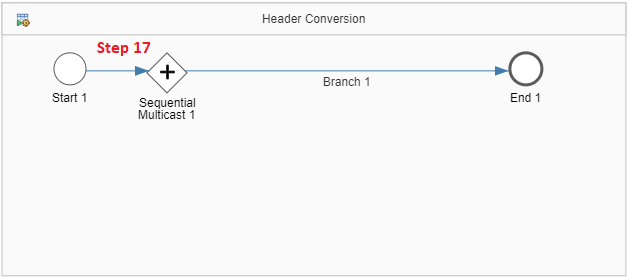


Process call will call the Local Integration process which will run a sub process (Header Conversion)



So, till now we have split message with grouping as 1 and sending it to the Local Integration Process (LIP) to convert Header, Body and Footer to CSV format.

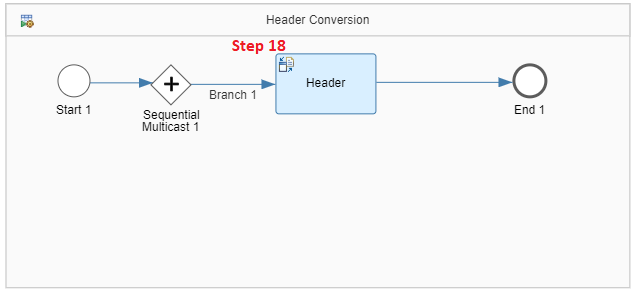
Step 17: In the LIP Header Conversion, Connect the Start 1 with sequential multicast from the pallet.



*Why Sequential Multi-cast?*

*We need the message divided into Header, Body and Foot as sequence. If we use parallel Multicast, message will be transmitter to all the branches at a time, whereas, sequential multicast, will send the message one followed by another in the defined sequence*

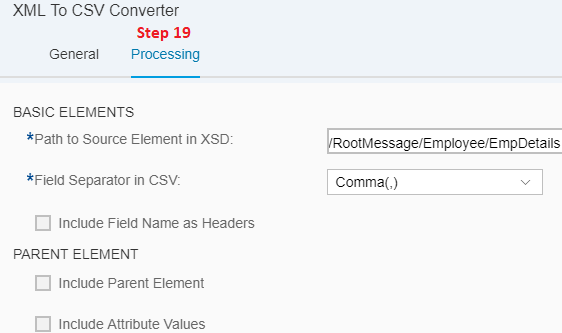
Step 18: Now, place the XML to CSV convertor (Header) between Sequential Multicast 1 and End 1. Rename it to Header.



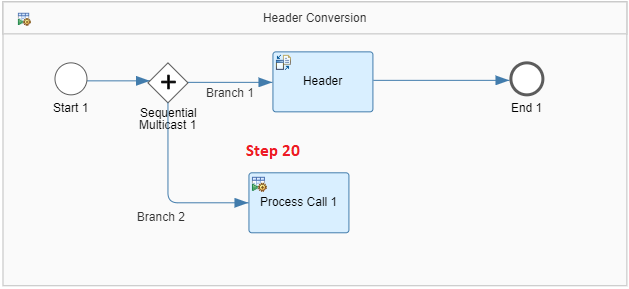
Step 19: Enter the processing details as per the below Screenshot

Path to Source Element in XSD: /RootMessage/Employee/EmpDetails

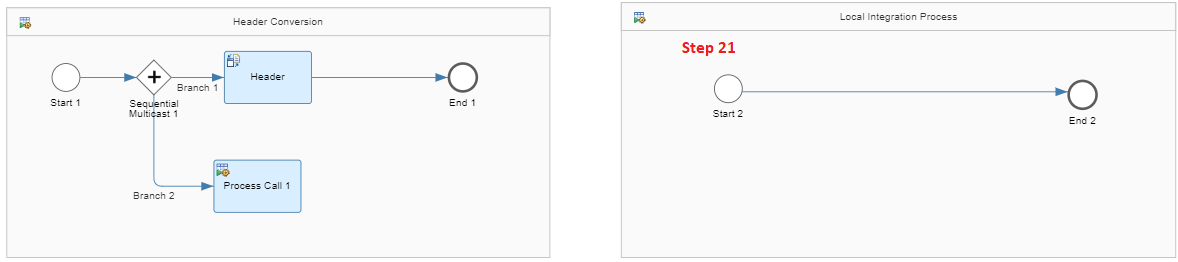
Field Separator in CSV: Comma (,)



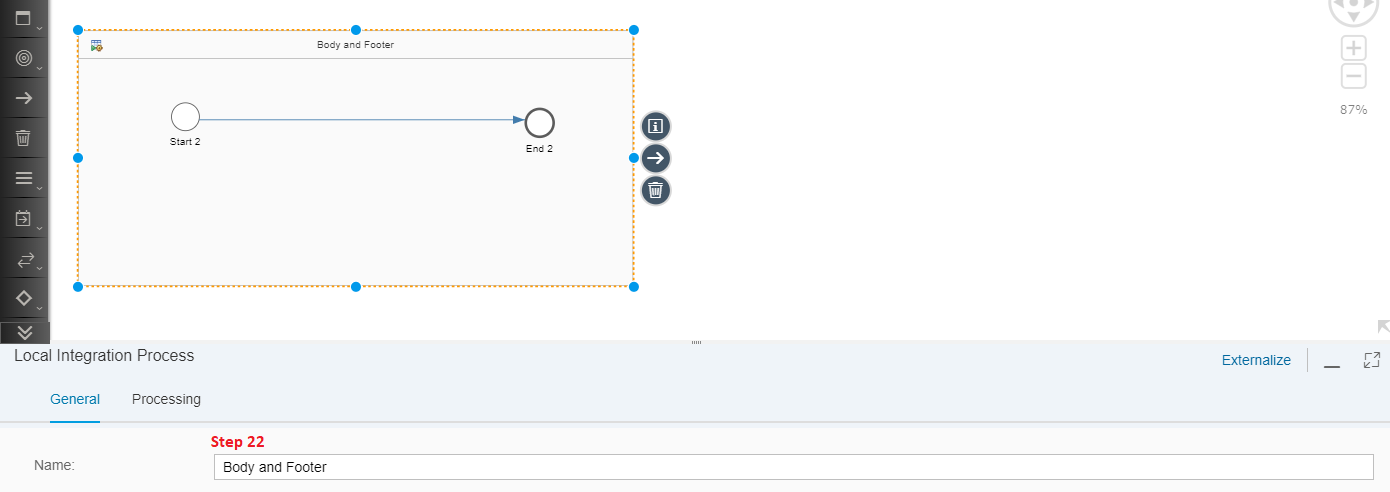
Step 20: From the pallet choose Call ---> Local Call ---> Process Call and place it in the Local Integration Process (Header Conversion) and connect branch 2 of Sequential Multicast with it.



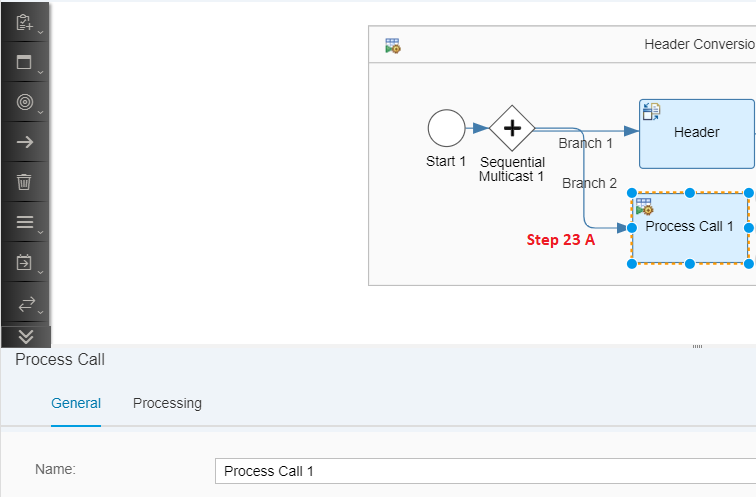
Step 21: Create a new local Integration process

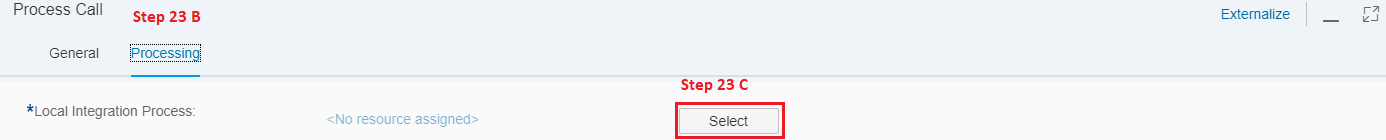


Step 22: Rename the Local Integration Process to Body and Footer



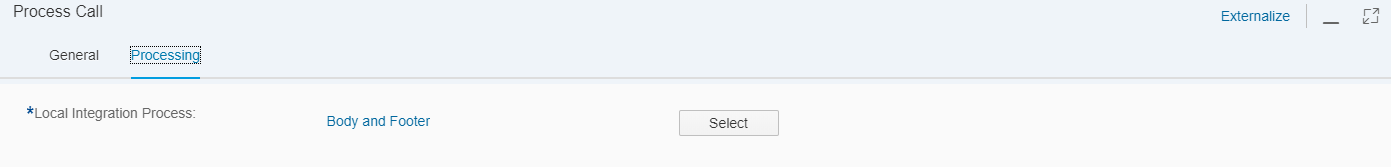
Step 23: Click on the Process call in LIP Header Conversion (23 A). Click on processing tab (23 B) and Select (23 C).





Step 24: Choose Body and Footer.

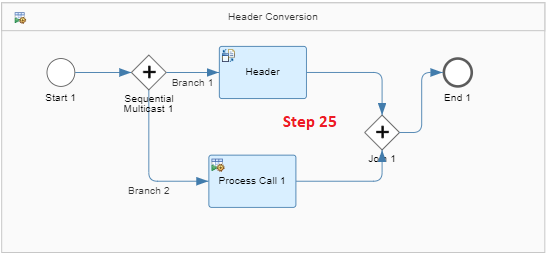




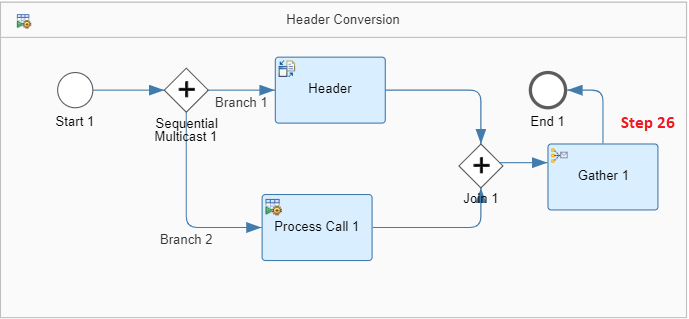
Step 25: In Local Integration Process Header Conversion, introduce Join from Pallet.

Pallet---> Routing---> Join. Connect Header (XML to CSV) and Join 1 and then Process call 1 and Join 1 **in sequence**.

*Join is used to combine multiple messages into a single target*



Step 26: Introduce Gather from pallets between Join 1 and End 1 in Local Integration Process Header Conversion. Pallet ---> Routing ---> Gather.

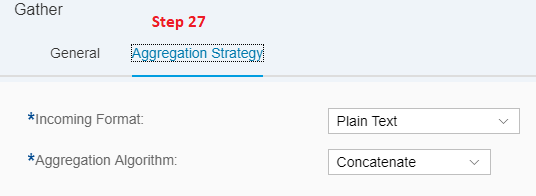


*Why Plain Text?*

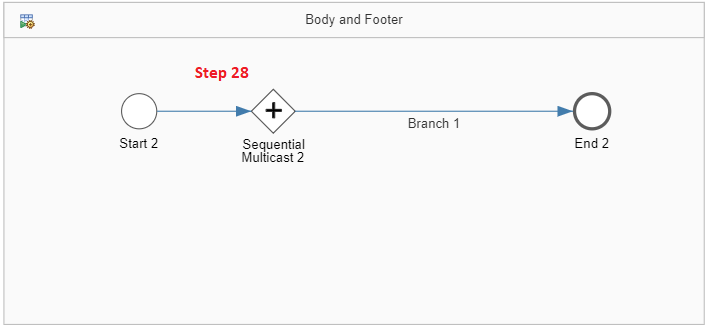
*Since the XML to CSV convertor converts the format from XML to CSV, usage of plain text becomes mandatory. Concatenate will bind the message as per the input sequence.*

*This completes the Flow for Local Integration Process 1 or Header Conversion. Now the next step is to proceed with Local Integration Process 2 or Body and Footer.*

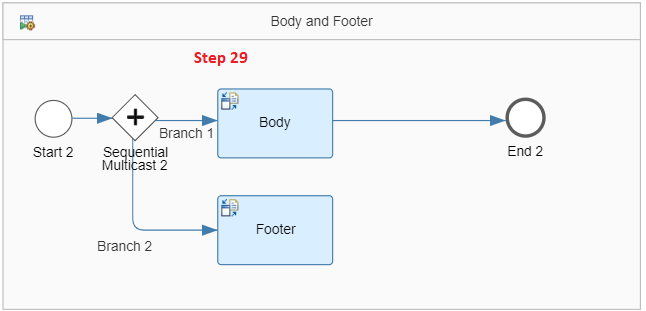
Step 27: Maintain the Aggregation strategy as shown below:



Step 28: Insert Sequential Multicast2 between Start 2 and End 2 of LIP Body and Footer.



Step 29: Introduce two XML to CSV convertors (rename them to Body and Footer) between Sequential Multicast 2 and End 2. Don’t forget to maintain the sequence. First branch from sequential Multicast should be pointing to Body and the second branch from sequential Multicast should be pointing to Footer.

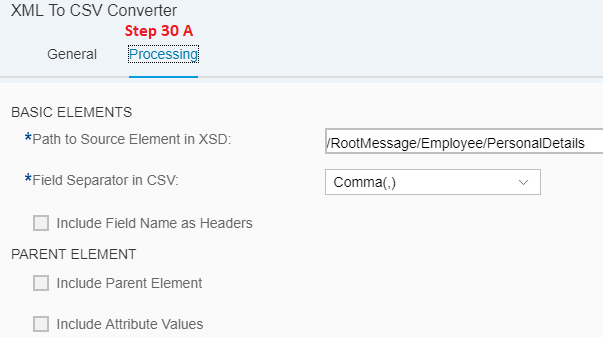


Step 30: Now configure the Body (30 A) and Footer (30 B) as show in below screenshots.

**Body**

Path to Source Element in XSD: /RootMessage/Employee/PersonalDetails

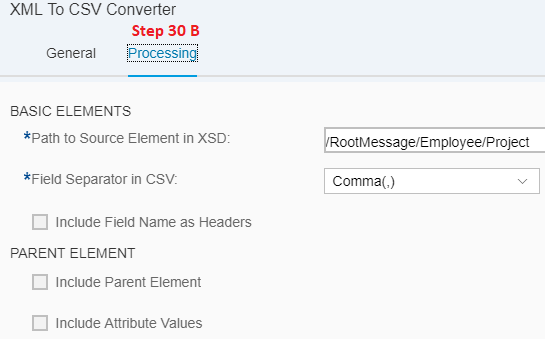
Field Separator in CSV: Comma (,)



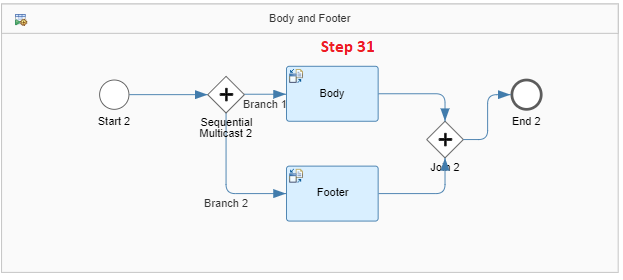
**Footer**

Path to Source Element in XSD: /RootMessage/Employee/Project

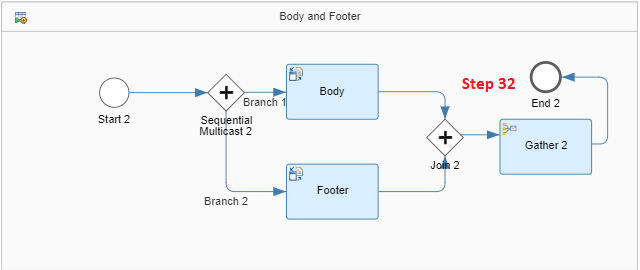
Field Separator in CSV: Comma (,)



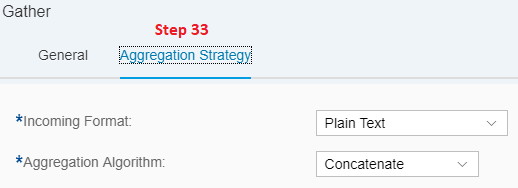
Step 31: Now join the Body and footer using Join function from Pallet. Pallet---> Routing---> Join. Remember the sequence must be followed. Body first and Footer second.



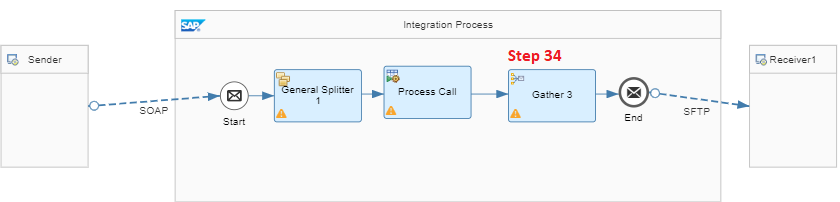
Step 32: Now Introduce the Gather function from Pallet to gather the message in sequence.



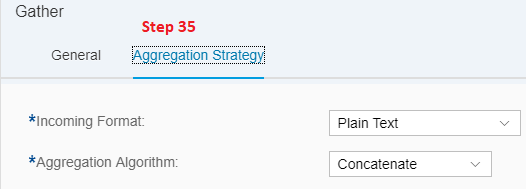
Step 33: Maintain Aggregation Strategy as shown below



Step 34 : Navigate to the Main Integration Process and place a Gather function between Process call and End Message

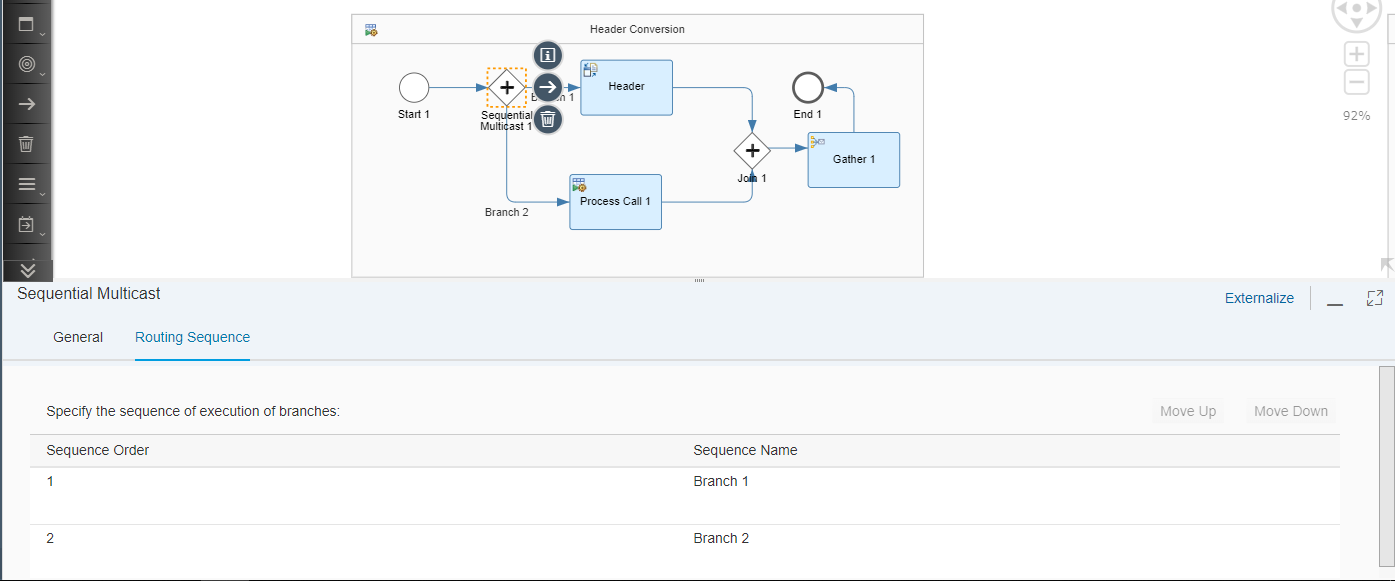


Step 35: Configure the Aggregation Strategy as shown below:

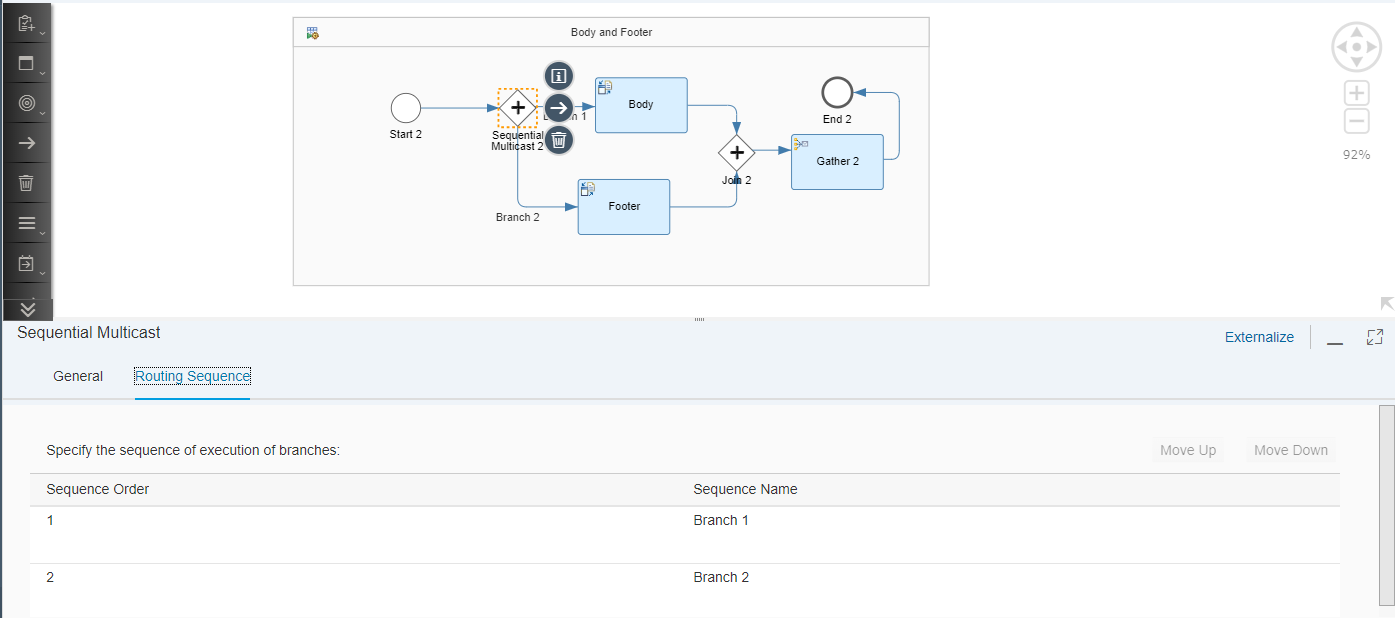


Check Points:

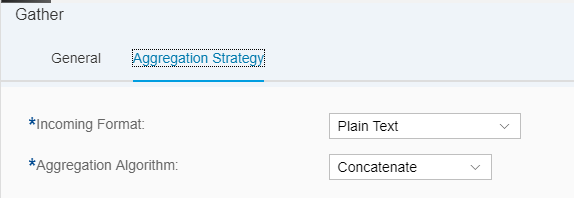
* Go to Sequential Multicast 1 in LIP (Local Integration Process) Header Conversion and check for the sequence.



* Go to Sequential Multicast 2 in LIP Body and Footer and check for the sequence.



* Check for the Aggregation Strategy in Gather.



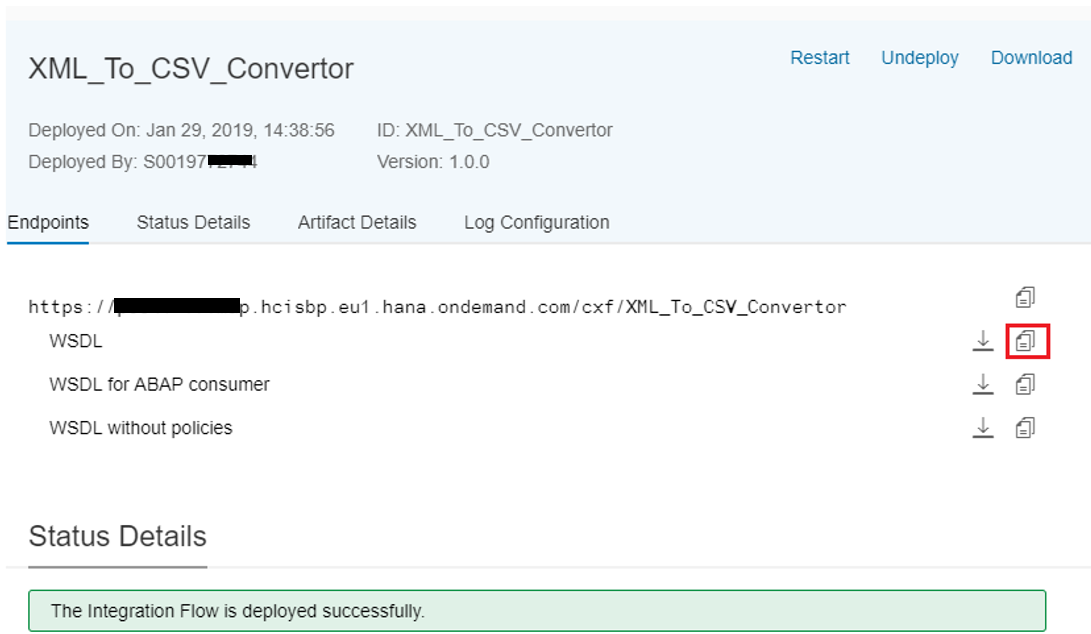
# **Testing:**

**Configure the SOAP UI for testing.**

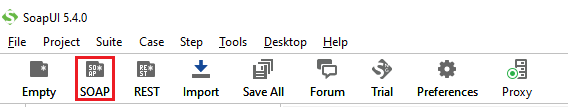
Step 1: Now we are done with the Setup. Next step is to trigger a message from SOAP UI to SAP ECC. For testing purpose we will use SOAP UI.

Step 2: Open SOAP UI

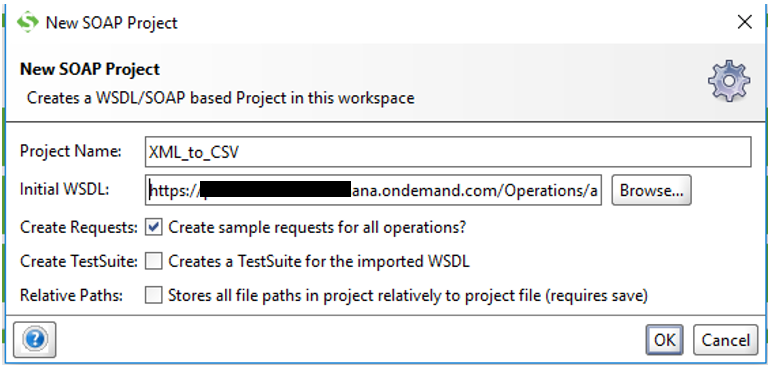
Step 3: Copy the proper WSDL file as per the below Screenshot



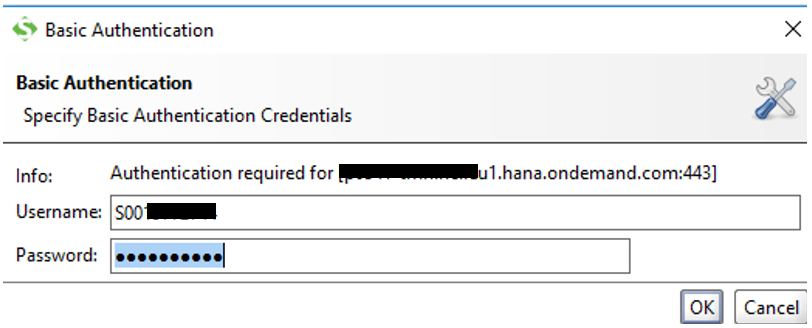
Step 4: Open New SOAP Project in SOAP UI.



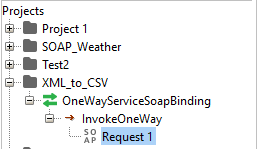
Step 5: Paste the Copied WSDL URL in Initial WSDL tab and click on OK



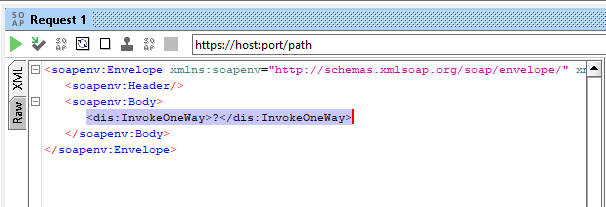
Step 6: Enter the CPI credentials and click on OK



Step 7: Expand XML\_to\_CSV and Double click on Request 1 in the Projects Navigator



Step 8: Delete the highlighted segment and put the XML payload



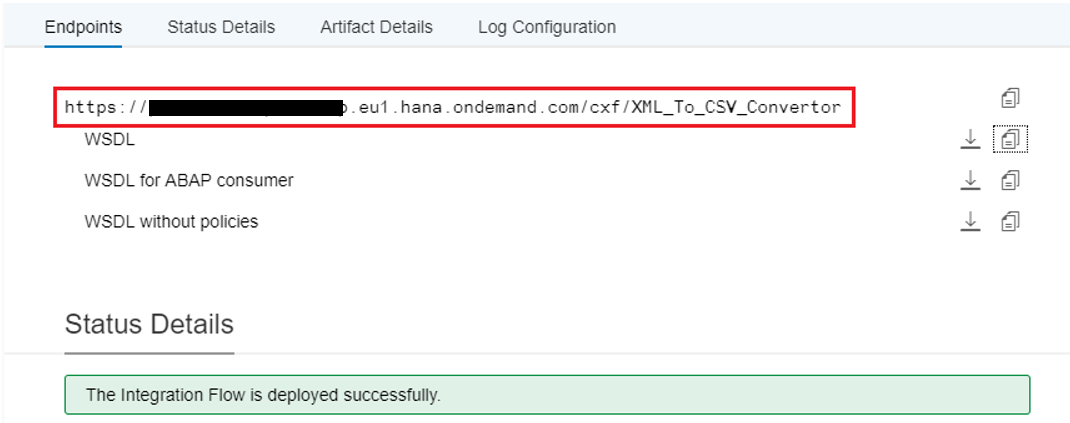


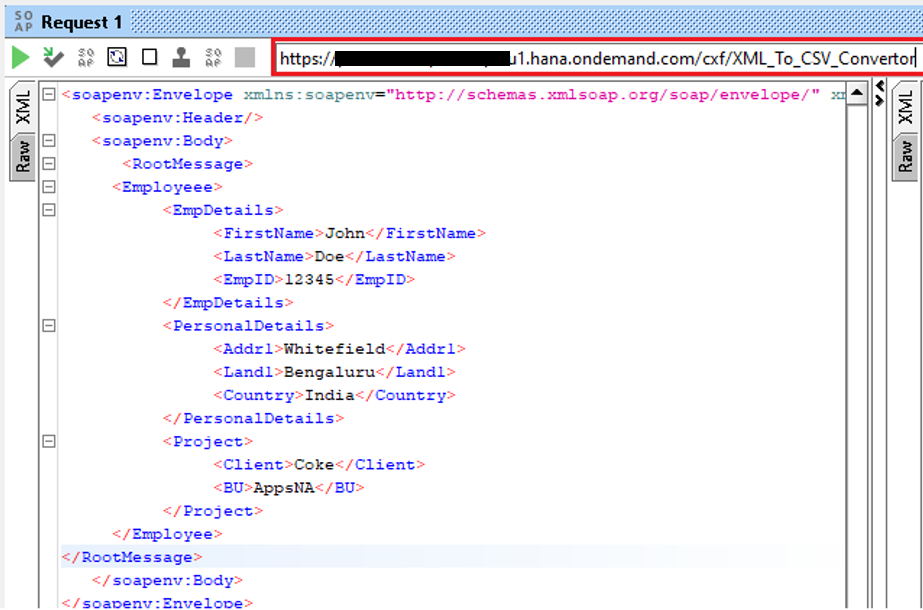
Payload attached below.

Sample Test File:

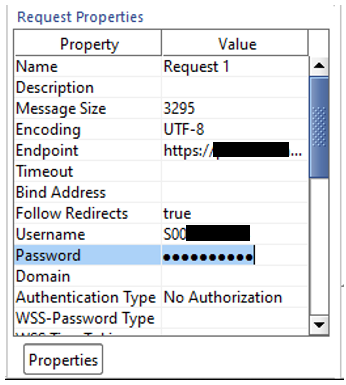


Step 9: Enter the End Point URL from the Monitoring Tab





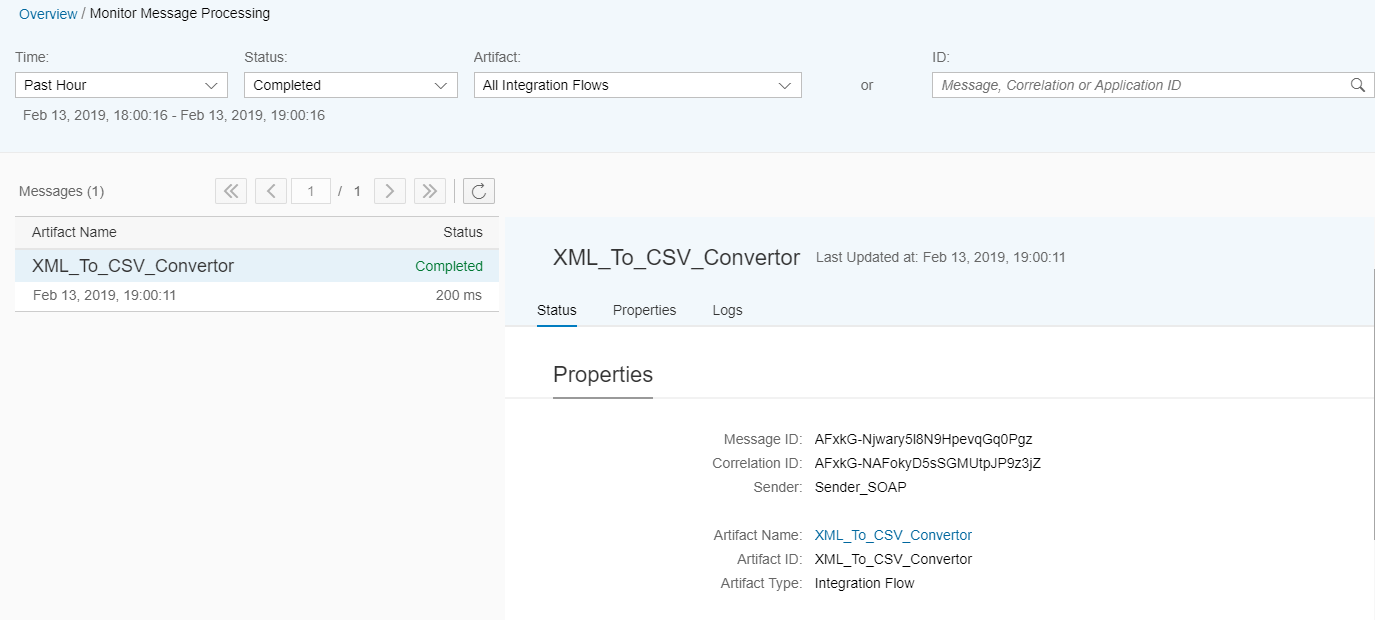
Step 10: Enter the USER Credentials in the properties tab of SOAP UI



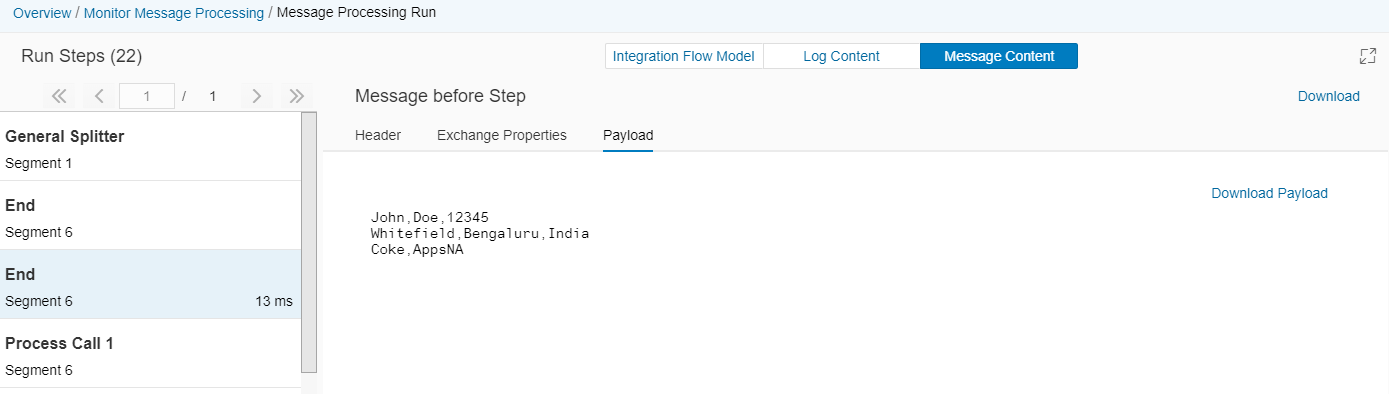
Step 11: Now we are good to trigger the message. Press the Start button in the Request1 tab



Step 12: Open Monitoring tab in CPI to view the message processing.



Step 13: Click on Logs ---> Trace---> Message Content-----> Payload



Bonus tip: we can check on every step of flow, if we click on Integration flow model

