

Lab 3 - IDoc Outbound Endpoint

Ver. 1.0.0

Overview

This next lab will focus on using the SAP Connector as an IDoc Outbound Endpoint. The flow that you create will send an IDoc to SAP to create a customer record.

Reminder: This lab requires the use of your own SAP instance. If you don't have one, you can leverage a hosted SAP service. For the purposes of this lab, we utilized [Sandbox SAP](#).

[Overview](#)

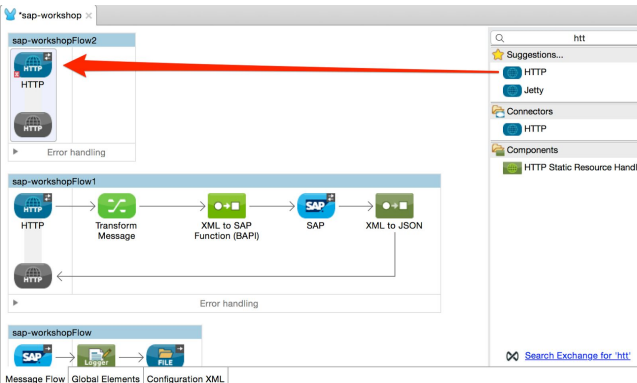
[Steps](#)

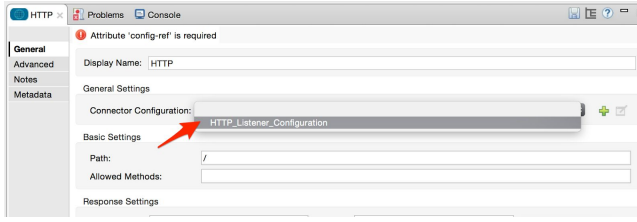
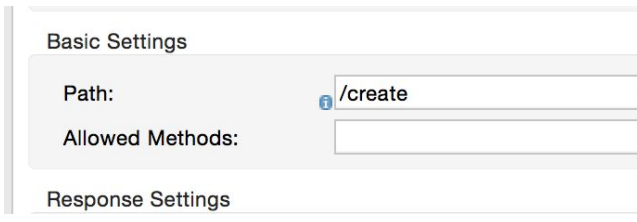
- [1. Setup HTTP Connector](#)
- [2. Configure SAP Connector](#)
- [3. Use DataWeave to Create an IDoc](#)
- [4. Transform XML to SAP IDoc](#)
- [5. Test Project](#)

[Summary](#)

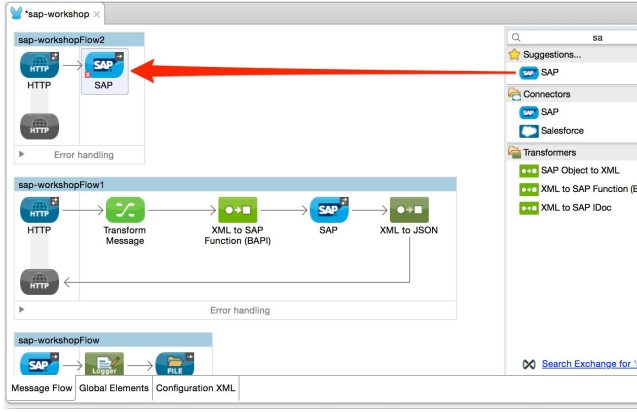
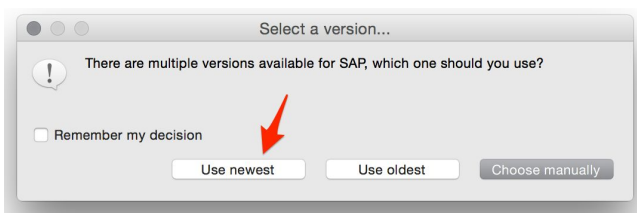
Steps

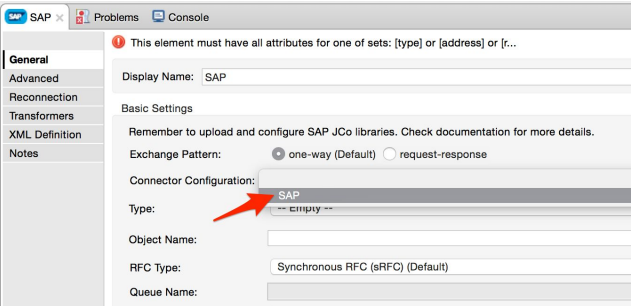
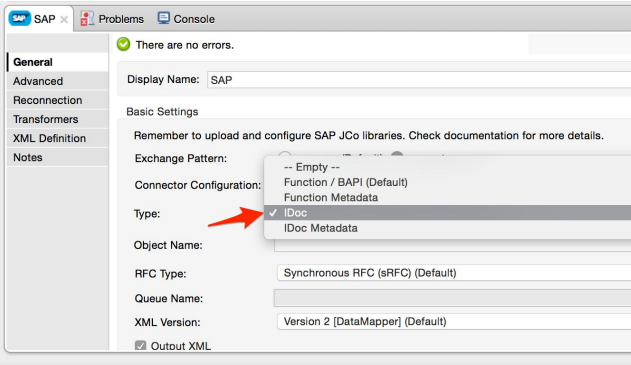
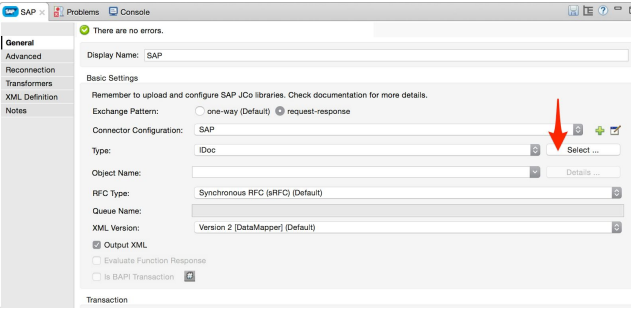
1. Setup HTTP Connector

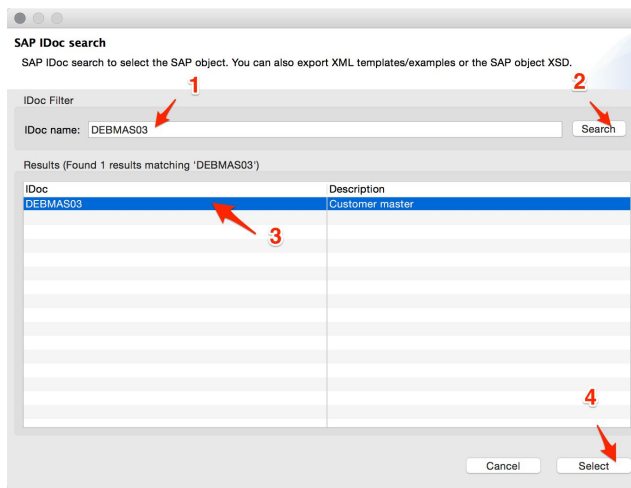
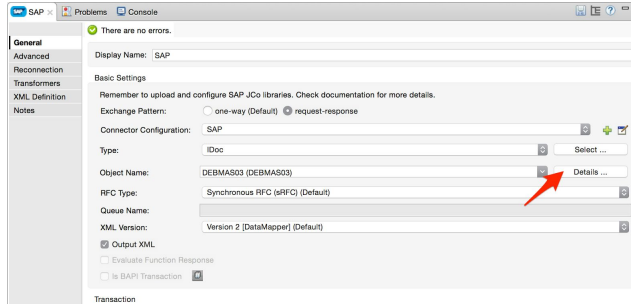
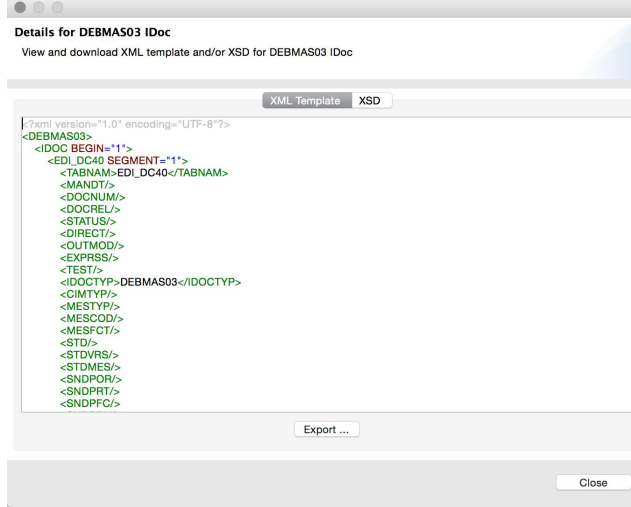
| | | |
|------------|---|--|
| <p>1.1</p> | <p>In the toolkit, search for the HTTP Connector by typing in 'http' in the Search bar.</p> <p>Drag and drop the HTTP Connector onto the canvas.</p> |  <p>The screenshot shows the MuleSoft IDE interface. On the right, a search bar contains the text 'http'. Below it, a list of suggestions includes 'HTTP' and 'Jetty'. A red arrow points from the 'HTTP' suggestion to the 'HTTP' connector icon in the 'sap-workshopFlow2' palette on the left. The main canvas shows a workflow diagram with an 'HTTP' connector at the start, followed by a 'Transform Message' step, an 'XML to SAP Function (BAPI)' step, an 'SAP' connector, and an 'XML to JSON' step. The workflow is titled 'sap-workshopFlow1'.</p> |
|------------|---|--|

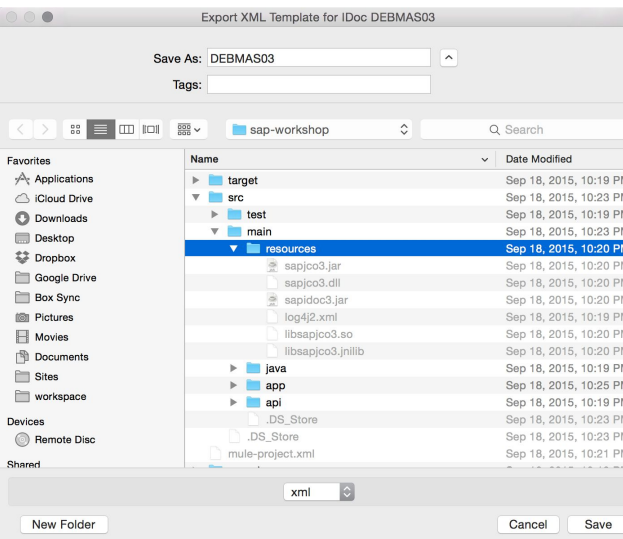
| | | |
|-----|--|--|
| 1.2 | In the Mule Properties tab for the HTTP Connector , select the configuration that was setup in Lab 2. The default should be HTTP_Listener_Configuration |  |
| 1.3 | Under the Basic Settings section, set the Path field to the following /create When you run the application later, you will use the following URL: http://localhost:8081/create |  |

2. Configure SAP Connector

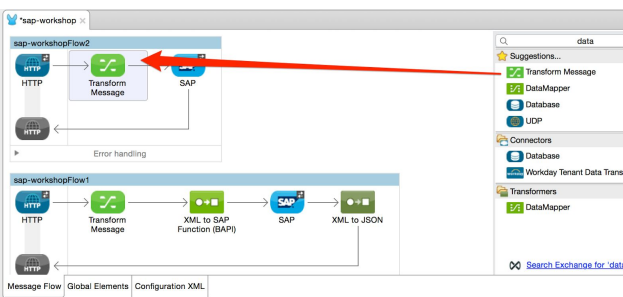
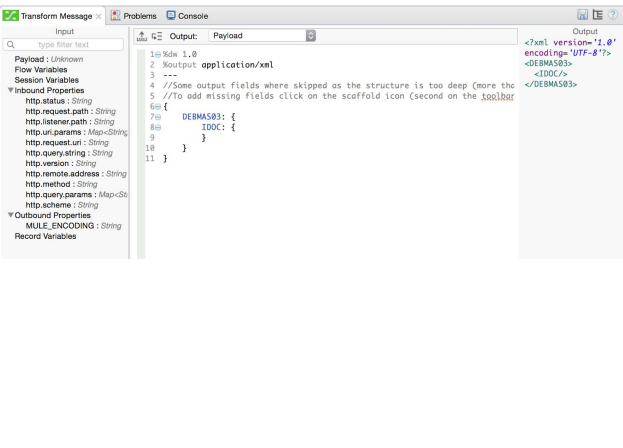
| | | |
|-----|---|--|
| 2.1 | <p>Next, let's add the SAP Connector.</p> <p>In the toolkit, search for the SAP Connector by typing in 'SAP' in the Search bar.</p> <p>Drag and drop the SAP Connector onto the canvas.</p> <p>This will set the SAP Connector as an outbound endpoint.</p> |  |
| 2.2 | If the Select a version... window pops up, select Use newest |  |

| | | |
|-----|--|--|
| 2.3 | In the Connector Configuration field, select the configuration that we created in Lab 1. It should be listed as SAP |  |
| 2.4 | Next, select request-response for the Exchange Pattern field. | <p>Remember to upload and configure SAP JCo libraries. Check documentation for more details.</p> <p>Exchange Pattern: <input type="radio"/> one-way (Default) <input checked="" type="radio"/> request-response</p> <p>Connector Configuration: SAP</p> |
| 2.5 | For the Type field, select IDoc |  |
| 2.6 | On the same line, next to the Type field, click on the Select button to search for the SAP object to be used in the transaction. |  |

| | | |
|-----|---|--|
| 2.7 | <p>Enter the following name in the IDoc name field:</p> <p>DEBMA503</p> <p>and click on Search.</p> <p>Select the result that is returned and click on Select</p> |  |
| 2.8 | <p>The SAP Connector provides a way to retrieve the XML and XSD for the SAP Object.</p> <p>Click on the Details button next to the Object Name field.</p> |  |
| 2.9 | <p>In the Details for DEBMA503 IDoc window, you can view and download the XML template and the XSD for the DEBMA503 IDoc.</p> |  |

| | | |
|------|---|--|
| 2.10 | Click on Export and save the XML Template to the src/main/resources folder. |  |
|------|---|--|

3. Use DataWeave to Create an IDoc

| | | |
|-----|--|---|
| 3.1 | <p>Now that the SAP Connector is configured with the correct credentials and SAP Object we intend to use, we need to setup the message to pass in.</p> <p>In the toolkit, search for the DataWeave component. Type in 'data' in the search and then drag and drop the Transform Message component into the canvas. Drop it between the HTTP Connector and the SAP Connector</p> |  |
| 3.2 | <p>In the Transform Message Configuration Properties window, you can see how the component scaffolds out the expected message to be passed to the SAP Connector using DataWeave</p> <p>The Output on the right shows a preview of the payload message as you modify the message using DataWeave</p> <p>More information about the DataWeave language can be found here:</p> <p>http://mulesoft.github.io/data-weave</p> |  |
| 3.3 | | <p>Let's make some modifications to the message. Copy the following script paste over the existing script in the Transform section of the DataWeave component. Be sure the change the items highlighted in red to match your SAP instance settings and customer record that you would like to insert.</p> |

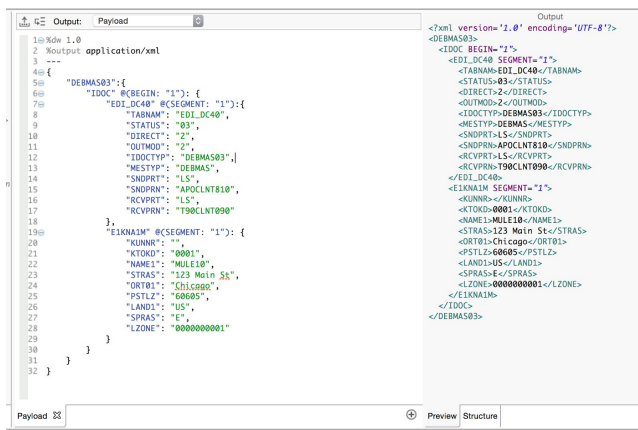
```

%dw 1.0
%output application/xml
---
{
    "DEBMAS03":{
        "IDOC" @(BEGIN: "1"): {
            "EDI_DC40" @(SEGMENT: "1"): {
                "TABNAM": "EDI_DC40",
                "STATUS": "03",
                "DIRECT": "2",
                "OUTMOD": "2",
                "IDOC TYP": "DEBMAS03",
                "MESTYP": "DEBMAS",
                "SNDPRT": "LS",
                "SNDPRN": "APOCLNT810",
                "RCVPRT": "LS",
                "RCVPRN": "T90CLNT090"
            },
            "E1KNA1M" @(SEGMENT: "1"): {
                "KUNNR": "",
                "KTOKD": "0001",
                "NAME1": "MULE10",
                "STRAS": "123 Main St",
                "ORT01": "Chicago",
                "PSTLZ": "60605",
                "LAND1": "US",
                "SPRAS": "E",
                "LZONE": "0000000001"
            }
        }
    }
}

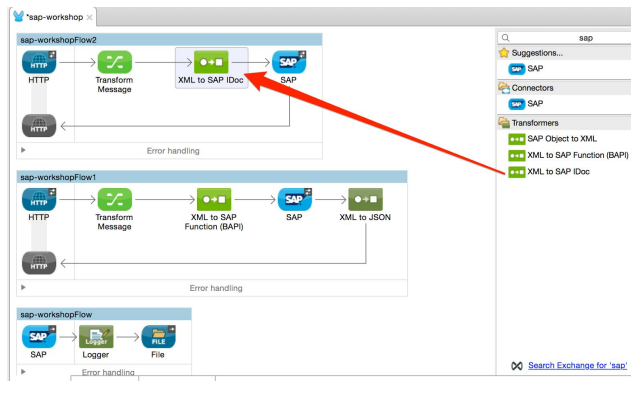
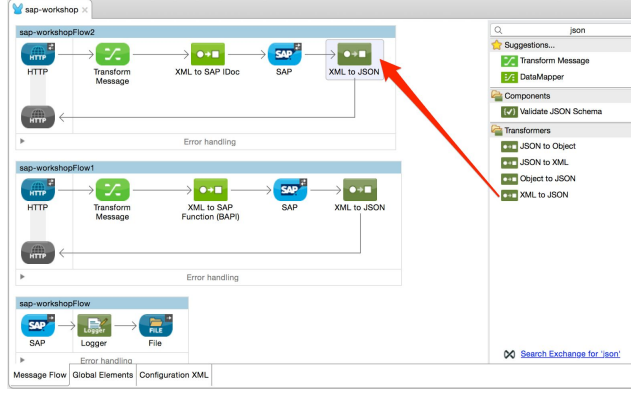
```

Notice how the preview on the right under **Output** updates the XML message that will be passed to SAP.

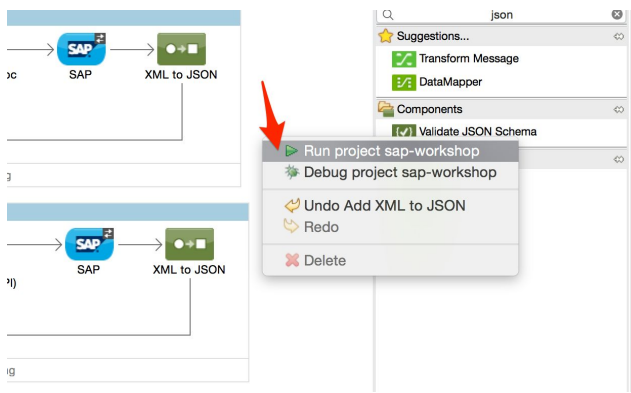
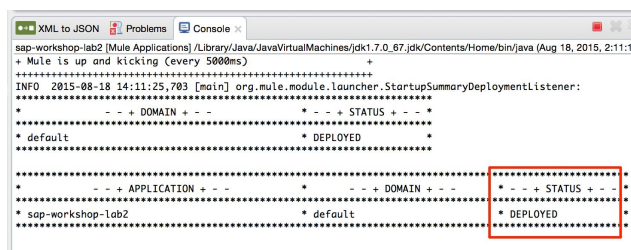
- 3.4 The script and output should look like the following screenshot.

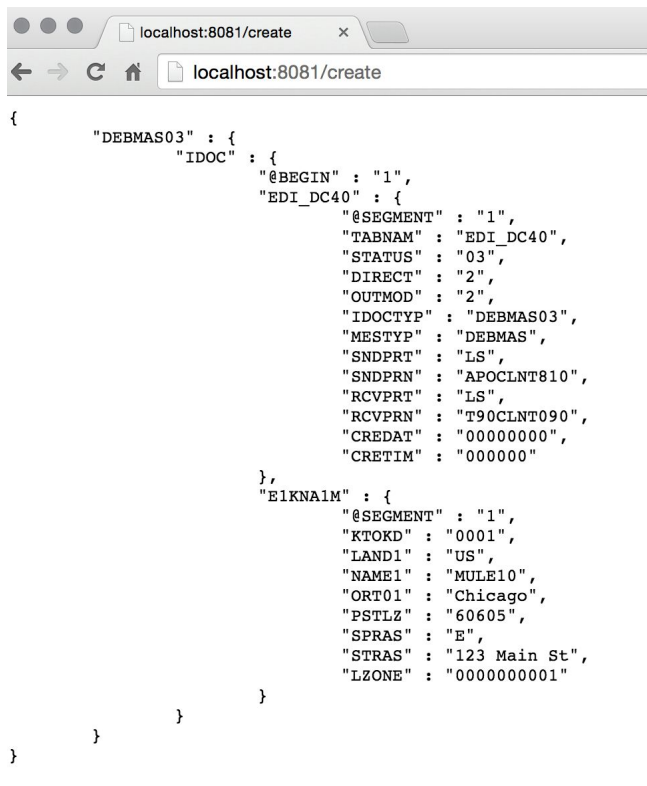
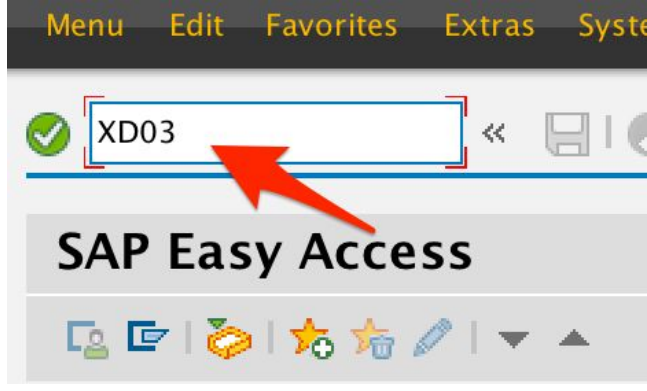

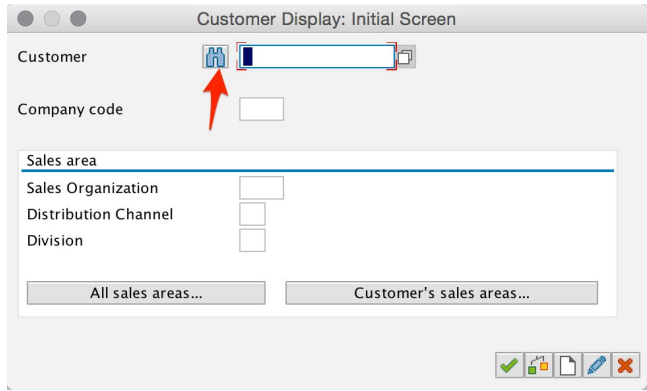


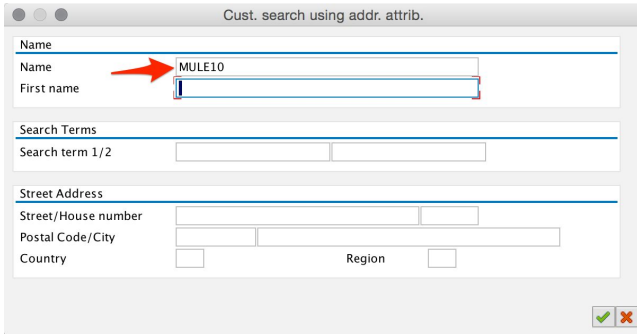
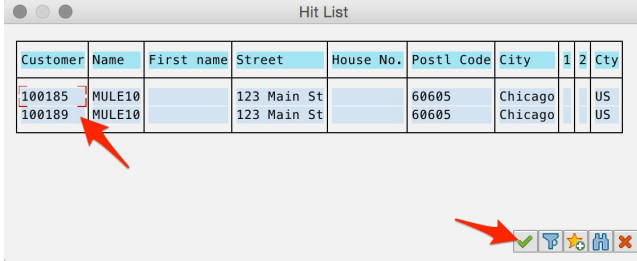
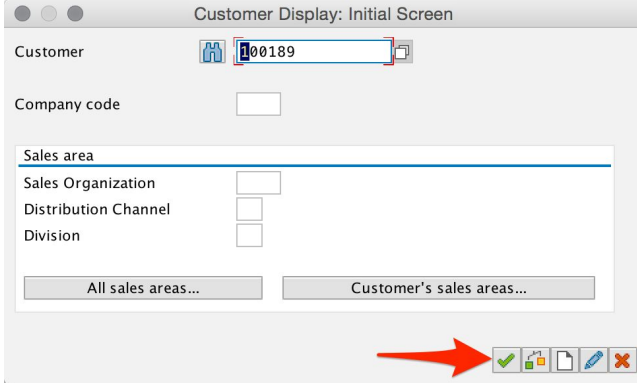
4. Transform XML to SAP IDoc

| | | |
|-----|--|---|
| 4.1 | <p>Now that the XML has the correct parameters, we need to transform it from XML to a SAP IDoc.</p> <p>From the toolkit, search for SAP and drag and drop the XML to SAP IDoc transformer and place it between Transform Message and the SAP Connector</p> |  |
| 4.2 | <p>Once the SAP Connector processes the function the data will be returned as XML. Let's go ahead and transform that and output the customer list as JSON data.</p> <p>Search for json and then drag and drop the XML to JSON transformer and place it after the SAP Connector.</p> |  |

5. Test Project

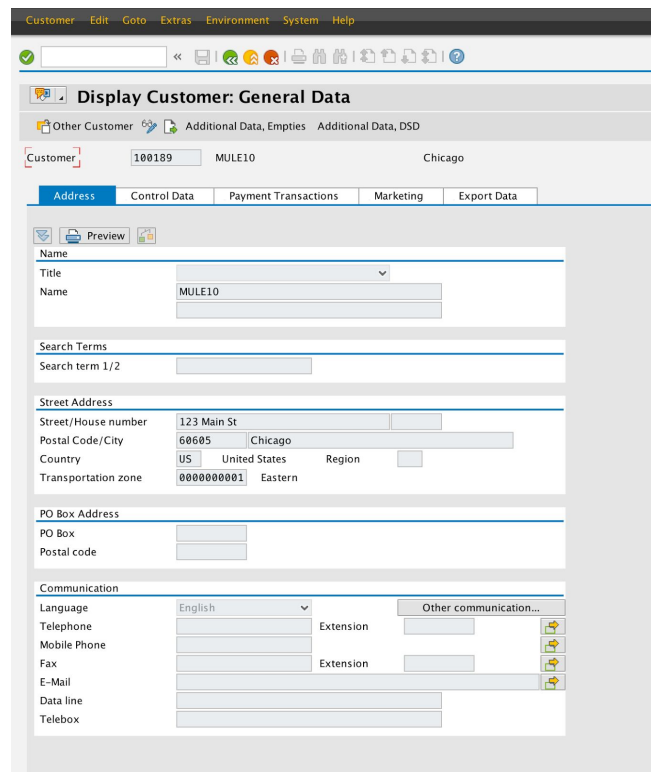
| | | |
|-----|---|--|
| 5.1 | <p>Our next step is to test the flow we've built.</p> <p>Right-click on the canvas and select Run project sap-workshop</p> |  |
| 5.2 | <p>The Console tab should pop-up now. Wait for the status to show DEPLOYED before moving onto the next step.</p> |  |

| | | |
|------------|---|---|
| <p>5.3</p> | <p>Let's test out our flow now. Switch to your browser and enter the following URL:</p> <p>http://localhost:8081/create</p> <p>If everything was configured correctly, you should see the following screen on the right.</p> |  <pre> { "DEBMAS03" : { "IDOC" : { "@BEGIN" : "1", "EDI_DC40" : { "@SEGMENT" : "1", "TABNAM" : "EDI_DC40", "STATUS" : "03", "DIRECT" : "2", "OUTMOD" : "2", "IDOCCTYP" : "DEBMAS03", "MESTYP" : "DEBMAS", "SNDPRT" : "LS", "SNDPRN" : "APOCLNT810", "RCVPRT" : "LS", "RCVPRN" : "T90CLNT090", "CREDAT" : "00000000", "CRETIM" : "000000" } }, "E1KNA1M" : { "@SEGMENT" : "1", "KTOKD" : "0001", "LAND1" : "US", "NAME1" : "MULE10", "ORT01" : "Chicago", "PSTLZ" : "60605", "SPRAS" : "E", "STRAS" : "123 Main St", "LZONE" : "0000000001" } } } </pre> |
| <p>5.4</p> | <p>Switch over to SAP GUI and enter in transaction code XD03 to find the customer that was inserted into SAP.</p> |  |
| <p>5.5</p> | <p>In the Customer Display: Initial Screen pop-up window, click on the Customer - Find  button.</p> |  |

| 5.6 | <p>In the Name field, enter the name of the customer that you passed over in the IDoc inside the node E1KNA1M > NAME1</p> |  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |
|----------|--|---|-------------|-----------|------------|---------|-----------|------------|------|---|---|-----|--------|--------|--|-------------|--|-------|---------|--|--|----|--------|--------|--|-------------|--|-------|---------|--|--|----|
| 5.7 | <p>The Hit List window should show the newly created customer record.</p> <p>Select the customer record and click on the Green check mark.</p> |  <table><thead><tr><th>Customer</th><th>Name</th><th>First name</th><th>Street</th><th>House No.</th><th>Postl Code</th><th>City</th><th>1</th><th>2</th><th>Cty</th></tr></thead><tbody><tr><td>100185</td><td>MULE10</td><td></td><td>123 Main St</td><td></td><td>60605</td><td>Chicago</td><td></td><td></td><td>US</td></tr><tr><td>100189</td><td>MULE10</td><td></td><td>123 Main St</td><td></td><td>60605</td><td>Chicago</td><td></td><td></td><td>US</td></tr></tbody></table> | Customer | Name | First name | Street | House No. | Postl Code | City | 1 | 2 | Cty | 100185 | MULE10 | | 123 Main St | | 60605 | Chicago | | | US | 100189 | MULE10 | | 123 Main St | | 60605 | Chicago | | | US |
| Customer | Name | First name | Street | House No. | Postl Code | City | 1 | 2 | Cty | | | | | | | | | | | | | | | | | | | | | | | |
| 100185 | MULE10 | | 123 Main St | | 60605 | Chicago | | | US | | | | | | | | | | | | | | | | | | | | | | | |
| 100189 | MULE10 | | 123 Main St | | 60605 | Chicago | | | US | | | | | | | | | | | | | | | | | | | | | | | |
| 5.8 | <p>Back in the Customer Display: Initial Screen window, click on the Green check mark to continue.</p> |  | | | | | | | | | | | | | | | | | | | | | | | | | | | | | | |

5.9

The customer that you inserted using an IDoc will be displayed if everything was configured successfully.



The screenshot shows the SAP 'Display Customer: General Data' screen. The menu bar at the top includes 'Customer', 'Edit', 'Goto', 'Extras', 'Environment', 'System', and 'Help'. Below the menu bar is a toolbar with various icons. The main header area displays 'Display Customer: General Data' and navigation links for 'Other Customer', 'Additional Data, Empties', and 'Additional Data, DSD'. The customer details are as follows:

- Customer:** 100189, MULE10, Chicago
- Address:** Control Data, Payment Transactions, Marketing, Export Data
- Name:** Title (dropdown), Name (MULE10)
- Search Terms:** Search term 1/2
- Street Address:** Street/House number (123 Main St), Postal Code/City (60605 | Chicago), Country (US | United States), Region, Transportation zone (000000001 | Eastern)
- PO Box Address:** PO Box, Postal code
- Communication:** Language (English), Telephone, Mobile Phone, Fax, E-Mail, Data line, Telebox, Extension, Other communication...

Summary

This final lab demonstrated how to build an IDoc and insert a customer into SAP using the SAP Connector and Anypoint Studio.