webMethods Flow Tutorial - No.6 Create a SEQUENCE Operation

Table of Contents [-]

Duration: 10 minutes

SEQUENCE operations are a means of grouping blocks of code that include processing variations. **SEQUENCE** is commonly used to combine processing variations to support graceful error-handling, similar to a try-catch block in Java. In this tutorial, you will add **SEQUENCE** steps to the customWriteToLog Flow service.

Prerequisites

This tutorial builds on concepts, techniques, and objects covered previously in:

- *1. Create an IS Package and Folders*
- *2. Create and Run a Flow Service*
- *3. Create Document Types*
- *4. Create a LOOP Operation*
- *5. Create a BRANCH Operation*
- The tutorials above must be completed or you can import the solution: Completed Export of 5. Create a BRANCH Operation.zip (http://techcommunity.softwareag.com/protected/download/developer-communities/webmethods/FreeTrial/Completed Export of 5. Create a BRANCH Operation.zip) using these instructions: **Import an IS Package** (http://softwareag.com/)
- The IS must be started. Instructions on how to start the IS are found in the **Prerequisites** part of the 1. Create an IS Package and Folders (http://techcommunity.softwareag.com/pwiki/-/wiki/Main/webMethods+Flow+Tutorial+-+Create+an+IS+Package+and+Folders).

Step Outline

You use **SEQUENCE** to implement a try-catch block by:

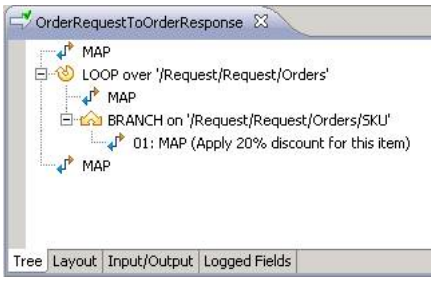
- Wrapping an existing service with **SEQUENCE** steps
- Verifying that service writes an error to the server log

Step 1: Add SEQUENCE Steps to the Service

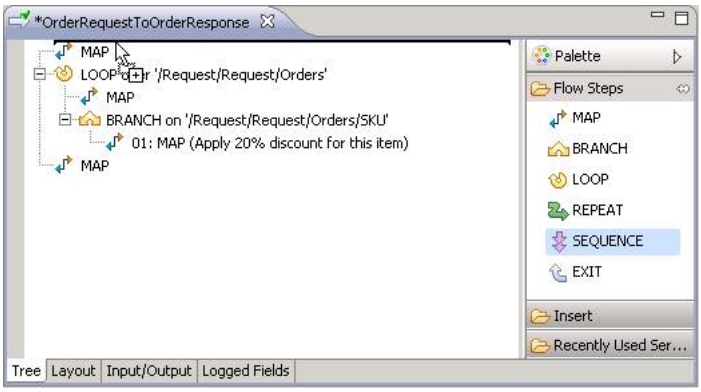
In this step: You will wrap the OrderRequestToOrderResponse Flow Service with **SEQUENCE** steps:

To wrap the service with **SEQUENCE** steps:

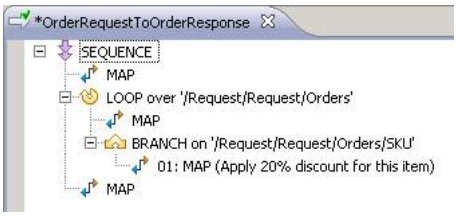
- Open the **FLOW_Tutorial.services:OrderRequestToOrderResponse** service:



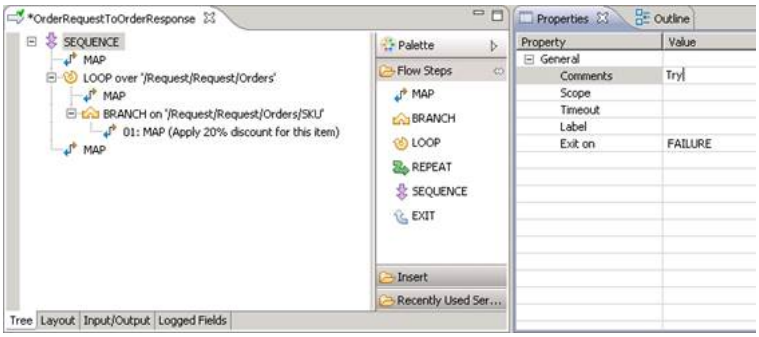
- Select the **SEQUENCE** tool to add a **SEQUENCE** step at the beginning of the service:



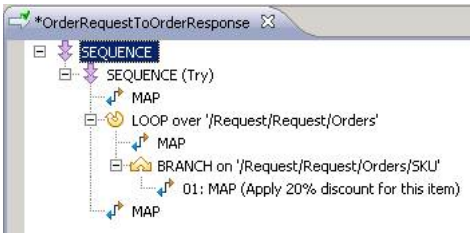
- Use the **Shift** toolbar button to indent all the steps under the initial **SEQUENCE** step:



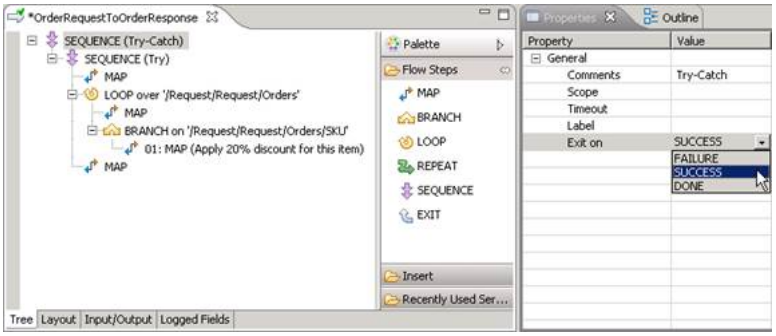
- Enter the text **Try** in the **SEQUENCE** Properties Comments field, and ensure that the **Exit on** field is set to **FAILURE:**



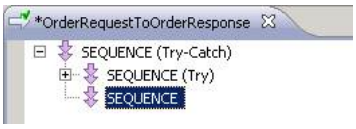
- Add a new **SEQUENCE** step as a parent of the **SEQUENCE(Try)** step:



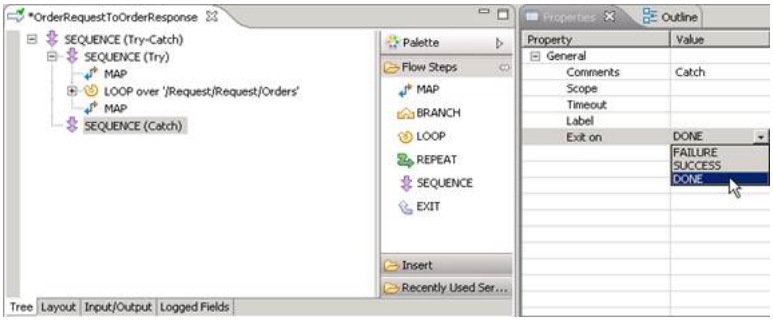
- Enter **Try-Catch** in the **Properties Comments** field, and enter **SUCCESS** in the **Exit** on field:



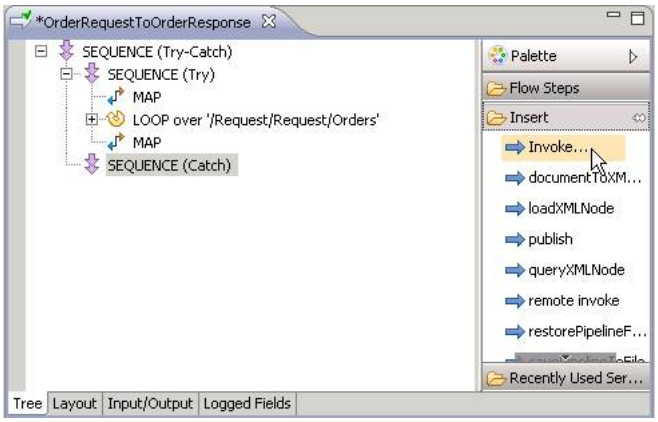
- Add a third **SEQUENCE** step directly after the **SEQUENCE(Try)** step as a child of the **SEQUENCE(Try-Catch)** step:

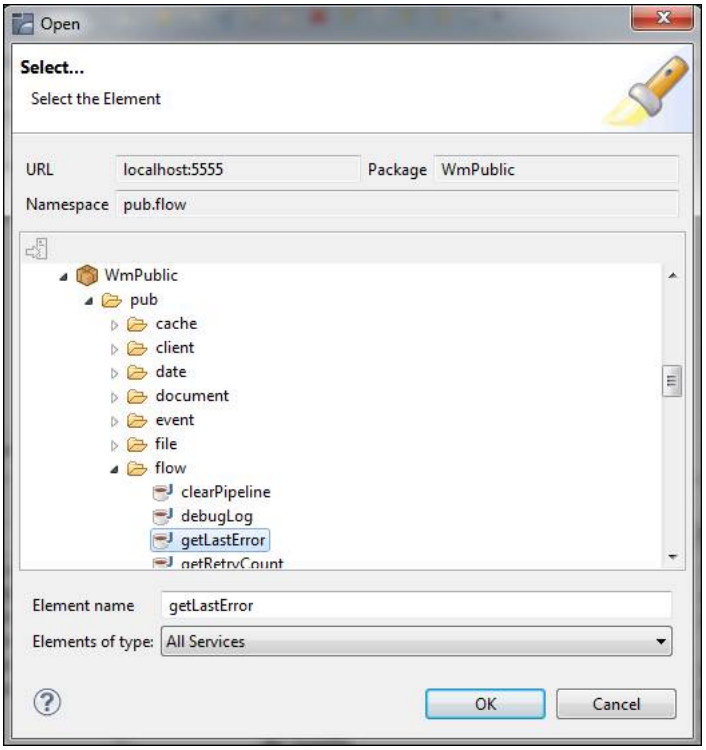


- Enter the text **Catch** in the **Comments** field, and enter **DONE** in the **Exit** on field:

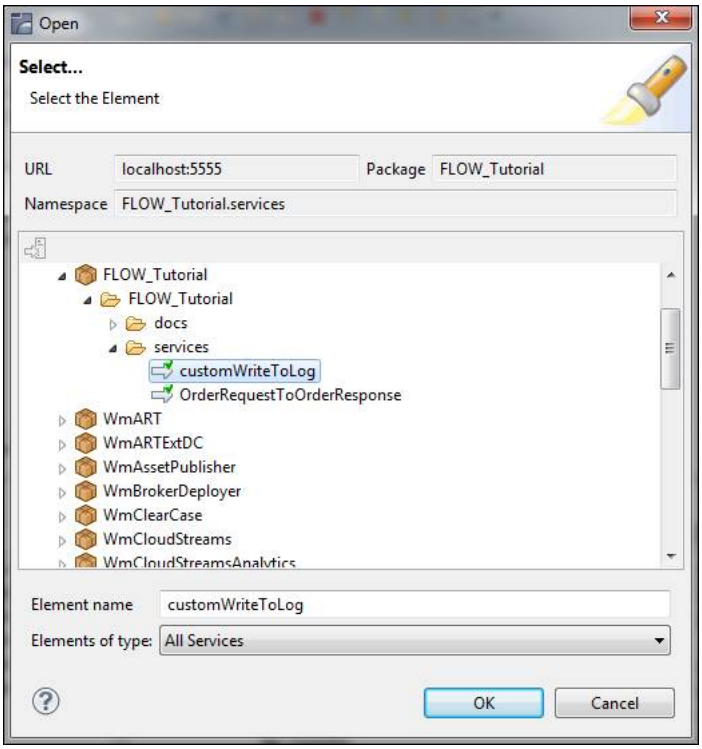


- Click and drag the **Invoke…** in the **Insert** part of the **Palette** and drop it on top of the **SEQUENCE(Catch)** step to add a call to the service **pub.flow:getLastError:**

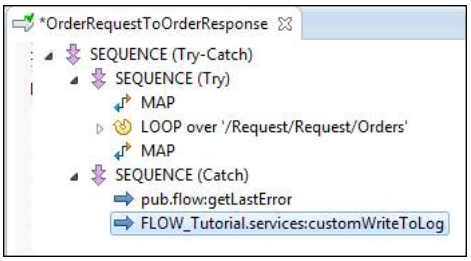




- Follow the same process to add a call to the service **FLOW_Tutorial.services:customWriteToLog** (the one you created in the first tutorial) after the call to **pub.flow:getLastError:**



Designer displays the calls in the flow service:

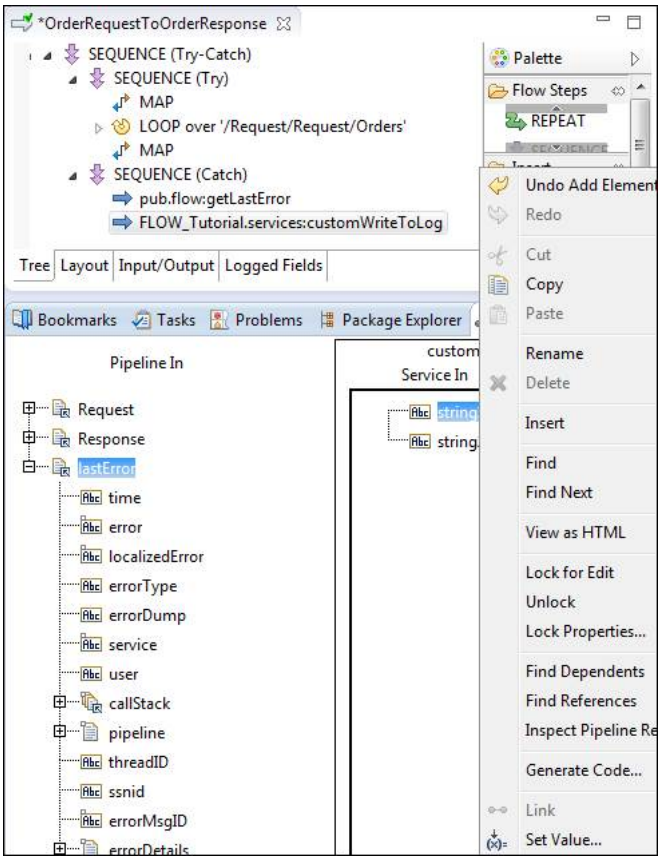


Note: Ensure that the calls are indented correctly so that they are children of the **SEQUENCE(Catch)** step.

You can now map the inputs for the **customWriteToLog** service.

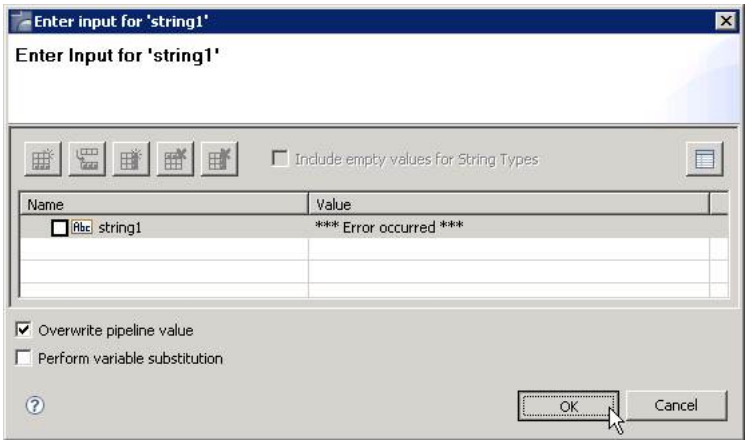
To map the inputs:

- Select the **customWriteToLog** step, then select the **Pipeline** tab
- Expand the lastError document reference in the **Pipeline In** area, and select the error string variable.
- Right-click on the **string1** node under **Service In** in the **customWriteToLog** area, and select **Set Value:**

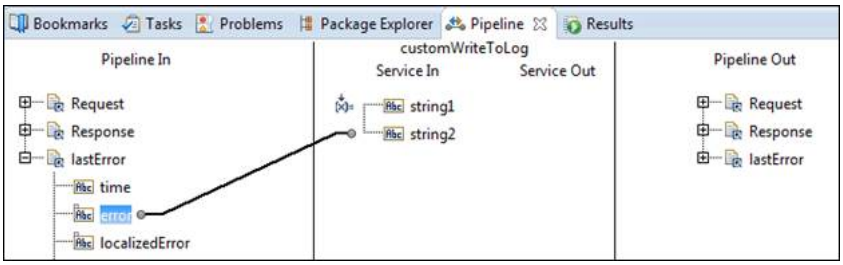


- Set the value of **string1** to:

* Error occurred *



- Link the **lastError/error** node to **string2:**



After you Save the service, you can test it in Designer.

Step 2: Load Input Data and Run the Service

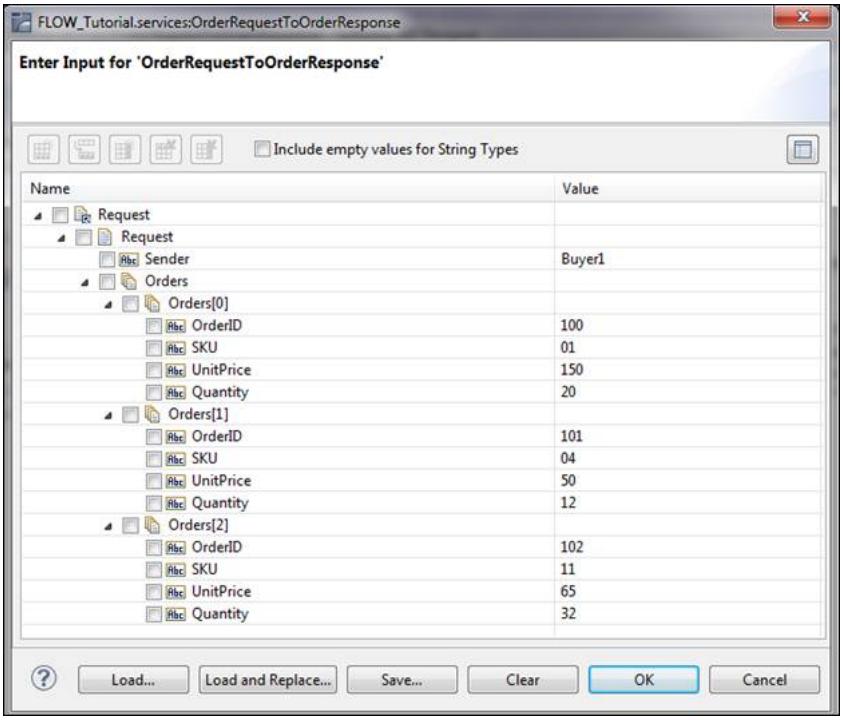
In this step: You will load a different input file and run the service.

You will run the service and input data by loading the following input file:

FLOW_Tutorial_6_Input.xml (http://techcommunity.softwareag.com/protected/download/developer-communities/webmethods/FreeTrial/FLOW_Tutorial_6_Input.xml)

To load the input file and run the service:

- Click the flow editor and select the arrow down part of the **Run** (
  
) menu from the Designer toolbar and choose to **Run As->Run Flow Service**



Designer displays the **Input** dialog.

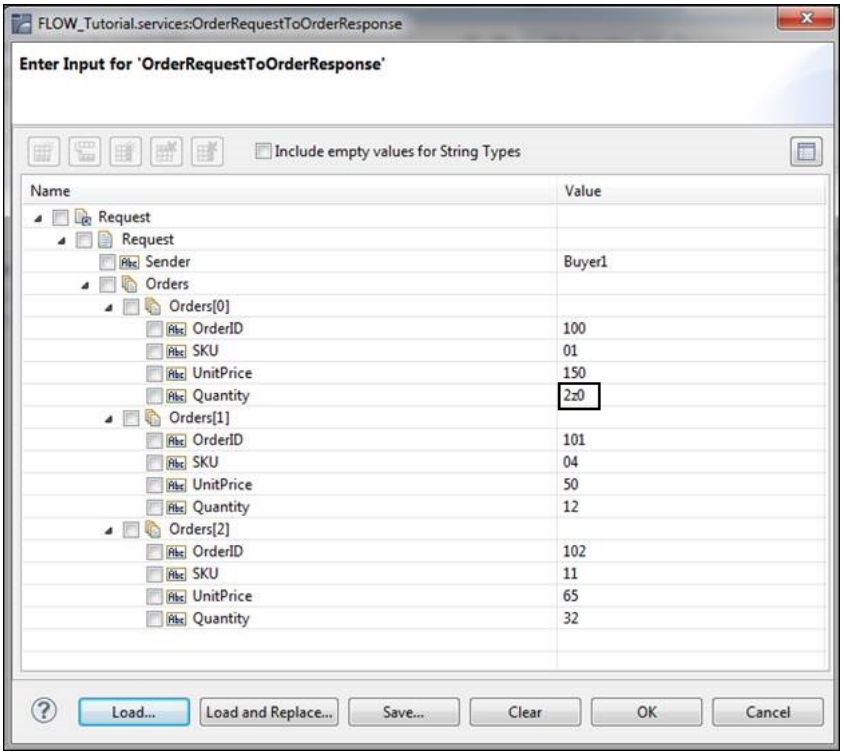
- Select **Load…**

Designer displays the **Open** file explorer.

- Locate and select the file:

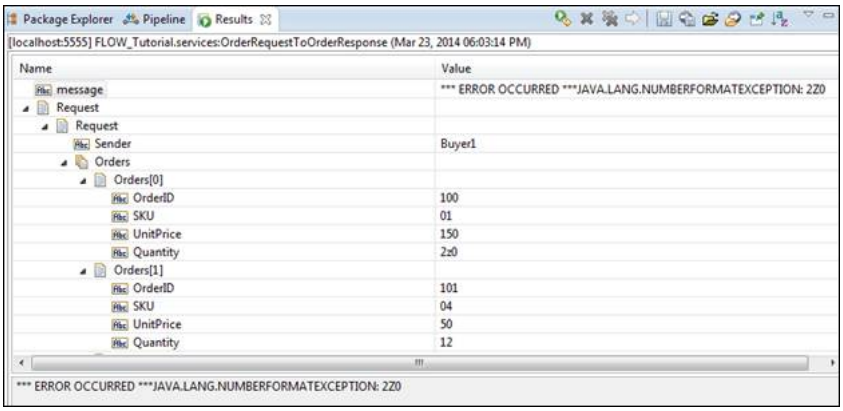
FLOW_Tutorial_6_Input.xml (http://softwareag.com/)

Designer displays the data loaded from the input file. Notice that the first Order in the input has an invalid non-numeric Quantity.



- Select **OK** to run the service

Designer runs the service and displays the **Results** view:



The error record shows an exception caused by the input **2z0**. The first order in the input file contains a value of **2z0** for quantity. The value **2z0** is not a valid number; consequently, the cost calculation operations failed.

You can now verify that the error is written to the server log file.

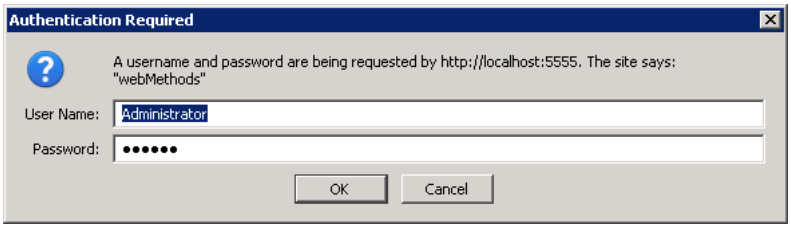
Step 3: Verify the Error in the Server Log

In this step: You will log into the integration server admin user interface to verify that the **CustomWriteToLog** service wrote the error to the log file.

To log into the integration server admin web user interface:

- Open the URL http://localhost:5555 (http://localhost:5555) in your browser of choice.

The Integration Server displays the Authentication Required dialog:



Use the following values to log into the Integration Server:

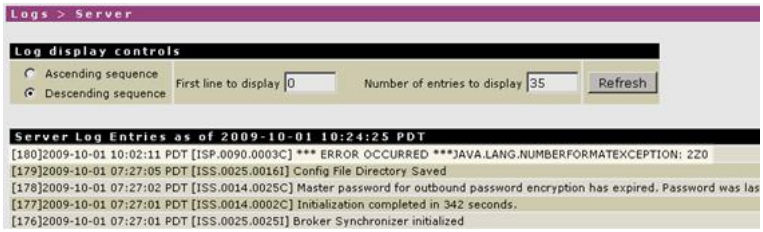
Field	Value
Username:	Administrator
Password:	manage

Note: These values may be stored from previous sessions. In this case, simply select OK.

Firefox displays the Integration Server Administration main page.

- Select **Logs > Server** from the left menu:

The **Logs > Server** page also displays the error:



Conclusion