



Website : <http://global-techhub.blogspot.com>
Email : ameerg.eai@gmail.com

Tel Telephone : +91-7337539091

Integrate a Java Service in IBM BPM

Introduction

In this article, we will show you how to write Java integration components for business process applications in IBM Business Process Manager. Depending on the functional or non-functional requirements of the business process application that you are implementing, you may find that none of the out-of-the-box connectors provide the functionality you need. For such cases, you may want to consider writing a Java integration component.

Overview

To use a Java service in Process Designer, you first discover the Java class and then generate an external service from it. The external service contains operations with inputs and outputs based on the methods in the discovered Java class. When you discover a service, if a corresponding external service already exists in Process Designer, you can either overwrite the existing service or create a new one.

We need to follow the below steps to Integrate with Java Service.

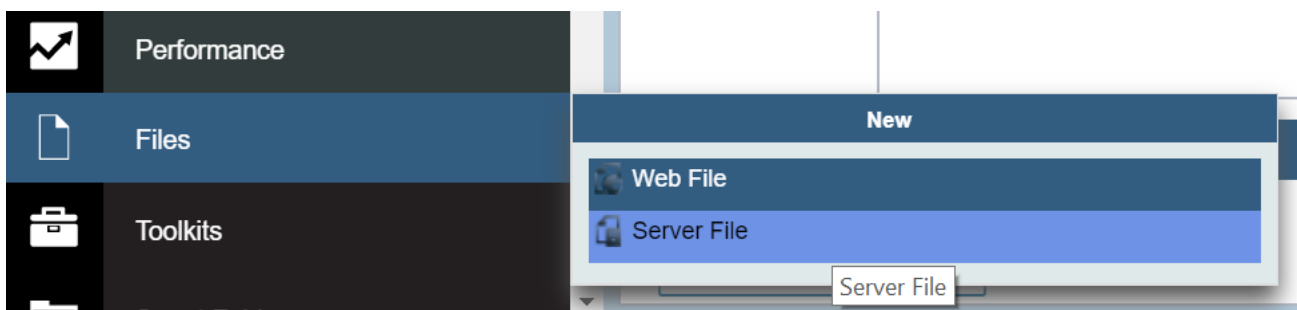
1. Import Java Application as jar.
2. Create a External Service.
3. Create a Service Flow.
4. Test the application

Import Java Application as jar:

Before creating the external service, add the JAR file that contains the classes that you need.

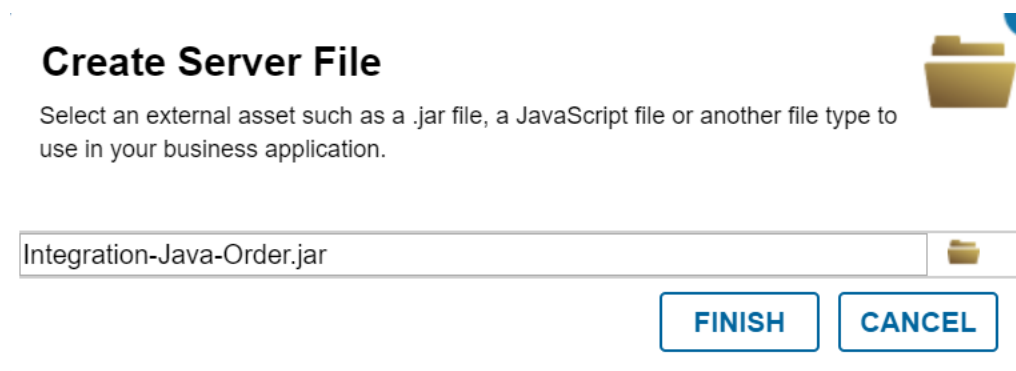
Open a process application in the **Designer view**.

Click the plus sign next to **Files** and select **Server File**.



In the New File window, click Browse to search for the file that you want.

Select the file, and then click **Finish**.



Create Server File

Select an external asset such as a .jar file, a JavaScript file or another file type to use in your business application.

Integration-Java-Order.jar

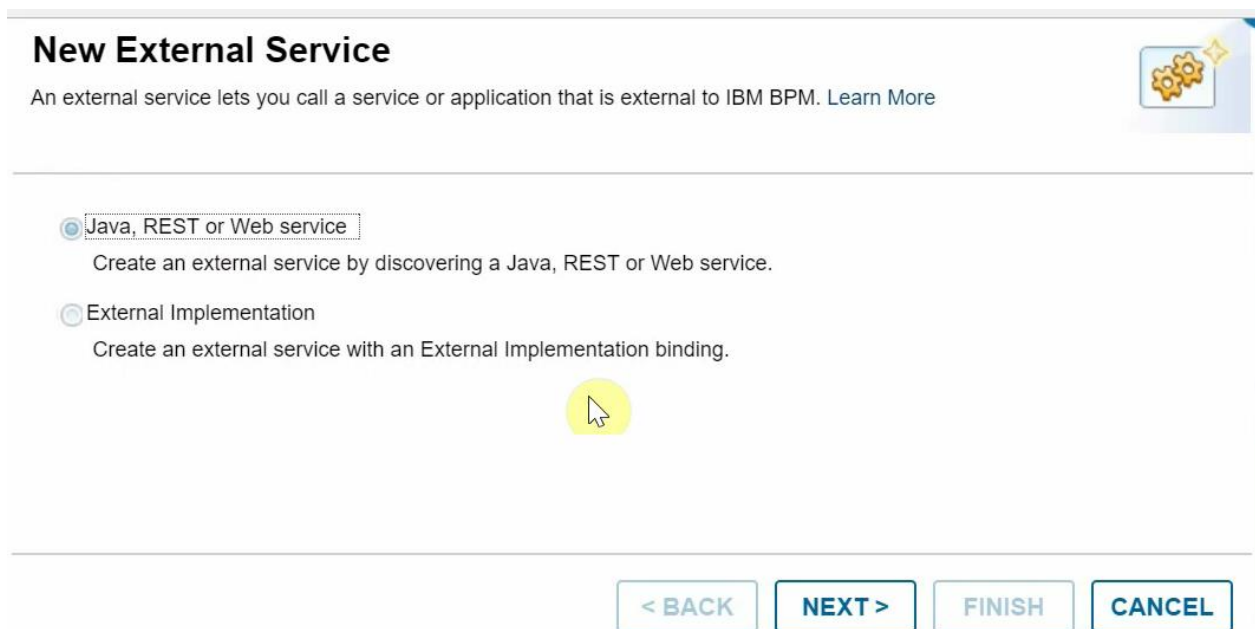
FINISH **CANCEL**

This screenshot shows a 'Create Server File' dialog box. It has a title bar, a main heading 'Create Server File', and a descriptive text. Below the text is a text input field containing 'Integration-Java-Order.jar'. To the right of the input field is a small folder icon. At the bottom right are two buttons: 'FINISH' and 'CANCEL'.

Create a External Service

Beside **Services** in the library navigation, click the plus sign (+). Select **External Service**.

In the New **External Service** page, choose **Java, REST or Web service** and click on next.



New External Service

An external service lets you call a service or application that is external to IBM BPM. [Learn More](#)

☒ **Java, REST or Web service**
Create an external service by discovering a Java, REST or Web service.

☐ **External Implementation**
Create an external service with an External Implementation binding.

< BACK **NEXT >** **FINISH** **CANCEL**

This screenshot shows the 'New External Service' page. It has a title bar, a main heading 'New External Service', and a descriptive text with a 'Learn More' link. Below the text are two radio button options: 'Java, REST or Web service' (which is selected) and 'External Implementation'. Each option has a brief description. At the bottom are four buttons: '< BACK', 'NEXT >', 'FINISH', and 'CANCEL'.

Select Browse server files (Java) as the method to discover a service.

Beside Managed file, click Select and select **Integration-Java-Order.jar**.

Beside Java class, select the '**OrderService**' class that you want from the drop-down list. Click Finish.


New External Service

An external service lets you call a service or application that is external to IBM BPM. [Learn More](#)

Select a method to discover the service.

Browse server files (Java) ▼

External service name:

Managed file:  [Integration-Java-Order.jar](#)

Java class: ▼

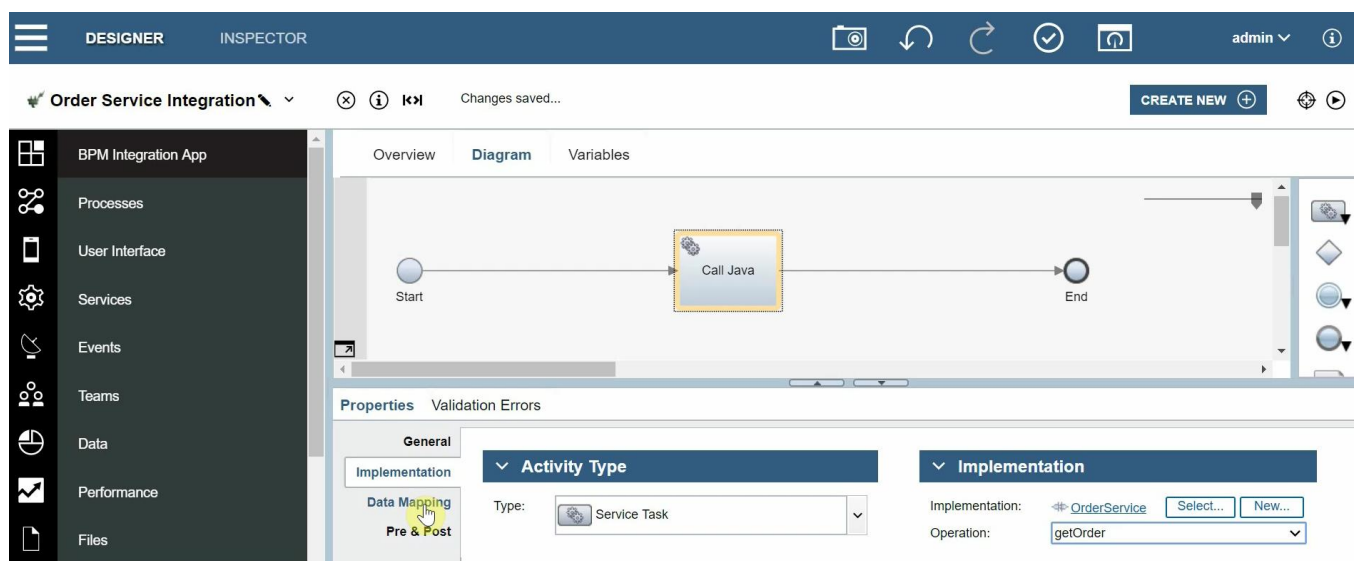
< BACK NEXT > FINISH CANCEL

Create a Service Flow

Beside **Services** in the library navigation, click the plus sign (+). Select **Service Flow**.

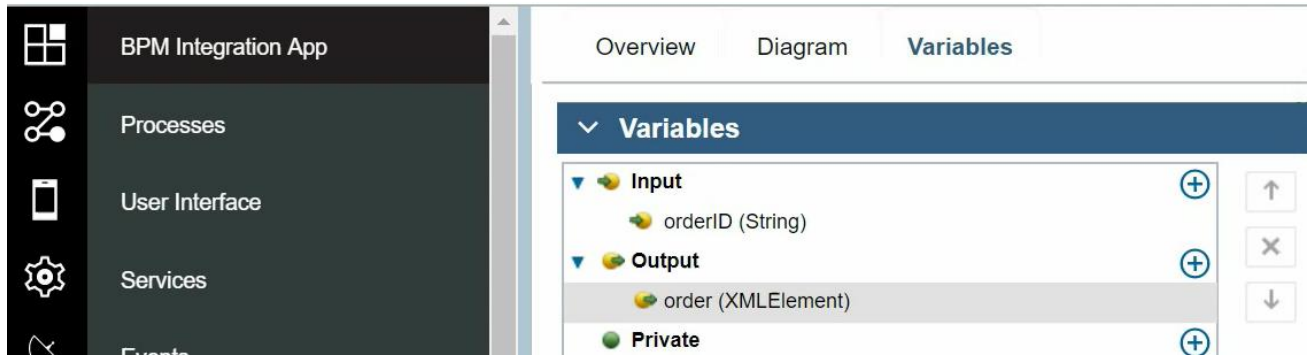
In the Service Flow editor, select a Service task. In the Implementation tab for the service, select the external Service which we have created in above step.

Select the operation name '**getOrder**' from drop down.

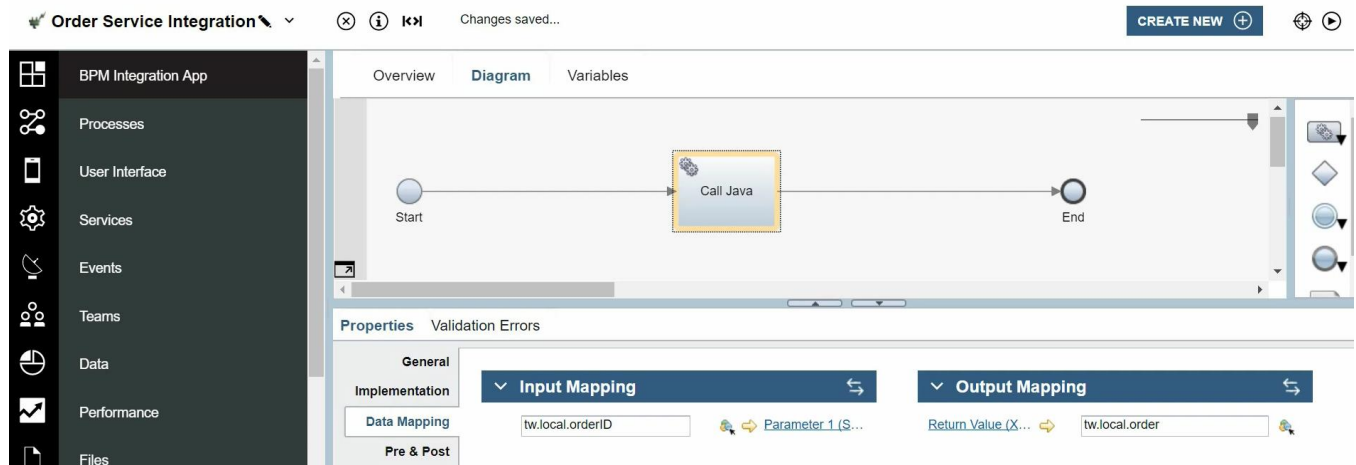


The screenshot shows the IBM BPM Designer interface. The top bar includes 'DESIGNER' and 'INSPECTOR' tabs, along with navigation icons and a user profile 'admin'. The main workspace displays a 'Service Flow' diagram with a 'Start' node, a 'Call Java' task, and an 'End' node. The left sidebar shows the 'BPM Integration App' navigation tree with 'Services' selected. The bottom panel shows the 'Properties' view for the 'Call Java' task. The 'Activity Type' is set to 'Service Task'. The 'Implementation' tab is active, showing the 'Implementation' dropdown set to 'OrderService' and the 'Operation' dropdown set to 'getOrder'.

Select the variable section and create input variable name as '**orderId**' type String and output variable name as '**order**' type 'XMLElement'.



Select Diagram and select service task → select properties and select Data Mapping and map the variables as shown below.

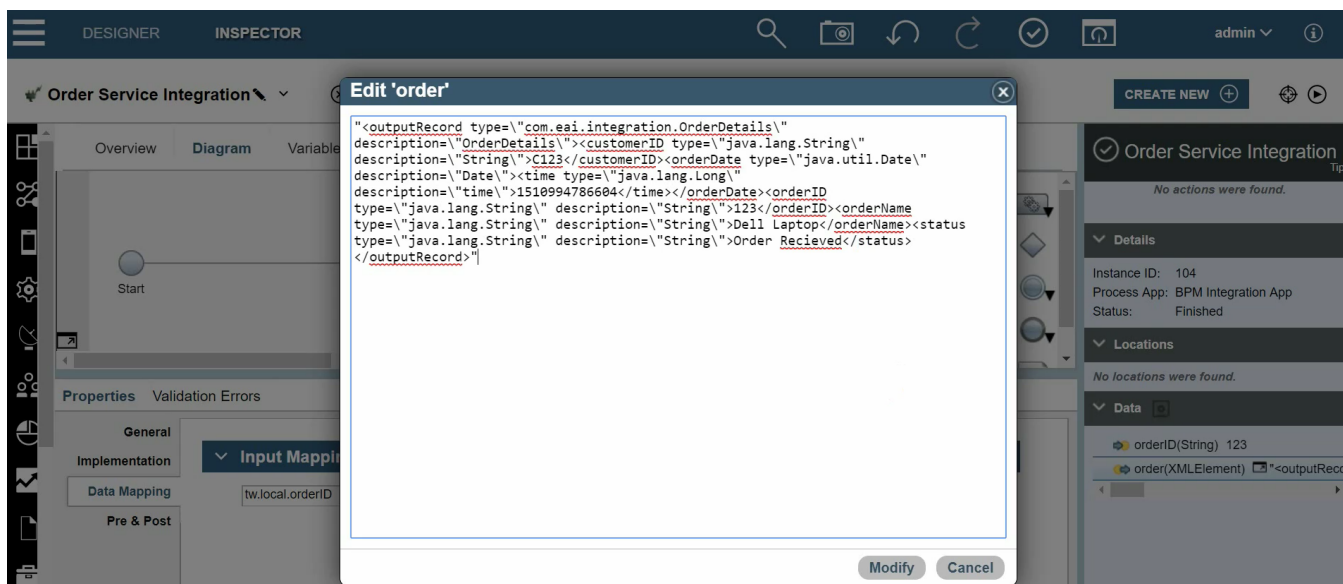


Test the application

We have implemented integration flow now and ready for testing.

Click on Debug button at right corner. Make sure in variables section you have selected default value for input 'orderId'.

Finally you can see response as shown below.



Conclusion

This article discussed the necessary building blocks for understanding what it takes to build and implement a Java integration component. You learned how variables defined in an IBM BPM process application are mapped to variables defined in Java. These are passed to a Java method and returned from a Java method to the process application in IBM BPM. The article also discussed the minimum set of classes required to develop the supporting code for a Java integration component for IBM BPM and the required dependencies to successfully compile your code. Using a Java integration component allows you to extend the functionality of your process application without too much effort.

Download

S. No	File Name	Size	Download
1	Integrate a Java service in IBM BPM.pdf	1.6 MB	Download
2	BPM_Integration_App_V1_Java Integration.zip	10 MB	Download

Get connected

