# Contents

**Chapter 1.** Introduction to WebSphere Adapter for IBM i - Running an RPG program tutorial................................................................. 1

- Learning objectives........................................................................................................ 1
- Audience......................................................................................................................... 1
- Software prerequisites.................................................................................................. 1

**Chapter 2.** Preparing to run through the tutorial ............................................ 2

- Configuration prerequisites.......................................................................................... 2
- Extracting the sample files......................................................................................... 2

**Chapter 3.** Outbound Processing – Running the RPG program...... 4

- Configuring the adapter for outbound processing ..................................................... 4
  - Creating the project..................................................................................................... 4
  - Setting connection properties for the external service wizard................................. 7
  - Generating business object definitions and related artifacts .................................. 8
- Deploying the module to the test environment............................................................ 13
- Testing the assembled adapter application................................................................. 16

**Chapter 4.** Introduction to Outbound/Inbound for Data Queues .... 1

- Learning objectives....................................................................................................... 1
- Audience....................................................................................................................... 1
- Software prerequisites................................................................................................. 2

**Chapter 5.** Preparing to run through the tutorial .............................. 1

- Configuration prerequisites......................................................................................... 1
- Extracting the sample files......................................................................................... 1

**Chapter 6.** Outbound Processing – Writing the message to (PutQueue) and reading the message from (GetQueue) the data queue. .......... 1

- Configuring the adapter for outbound processing ..................................................... 1
  - Creating the project..................................................................................................... 1
  - Setting connection properties for the external service wizard................................. 8
  - Generating business object definitions and related artifacts .................................. 9
- Deploying the module to the test environment............................................................ 15
- Testing the assembled adapter application................................................................. 17

**Chapter 7.** Inbound processing on a data queue............................ 25

- Configuring the adapter for inbound processing..................................................... 25
Chapter 1. Introduction to WebSphere Adapter for IBM i - Running an RPG program tutorial

WebSphere® Adapter for IBM® i V6.2 exchanges business data between system i and J2EE applications. The adapter retrieves from and writes to the data queue and runs RPG programs.

The document demonstrates the following scenario:

1. Running an RPG program with the help of Adapter for IBM i

This scenario demonstrates how WebSphere Adapter for IBM i V6.2 performs outbound operations.

Learning objectives

After completing this tutorial, you should be able to perform the following tasks:

Create an adapter project in WebSphere Integration Developer.

Discover services and associated business objects from the enterprise information system (EIS) and make them part of the adapter project.

Create a deployable module that you install on WebSphere Process Server or WebSphere Enterprise Service Bus.

Test the module and validate the results.

Audience

This tutorial is for integration developers who design, assemble, test, and deploy business integration solutions.

Software prerequisites

To use this tutorial, you must have the following applications installed:

WebSphere Integration Developer version 6.2

WebSphere Process Server version 6.2
Chapter 2. Preparing to run through the tutorial

Configuration prerequisites

Before trying any test based on this tutorial, complete the following tasks:

Create a sample RPG program on an IBM i system.

Compile the sample RPG program with the compiler parameter PGMINFO(*PCML) to generate a PCML file.

Extracting the sample files

Replicas of the artifacts that you create when using the external service wizard are provided as sample files for your reference. Use these files to verify that the files you create with the external service wizard are correct.

Note that the values for the i5/OS host name, user name, and password in the sample artifacts are from the IBM test lab. You need to change or set them appropriately for your environment.

Following table lists the artifacts that are shipped as part of samples which user can use to verify when running an RPG program.

1) RPG_Outbound_PM.zip – Project Interchange file which includes generated sample artifacts for running an RPG program

<table>
<thead>
<tr>
<th>File/artifact name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>add2num</td>
<td></td>
</tr>
<tr>
<td>add2num/ISeriesImport.import</td>
<td>Contains the SCA import for the resource adapter.</td>
</tr>
<tr>
<td>add2num/ADD2NUM.xsd</td>
<td>Business object definition for the RPG program</td>
</tr>
<tr>
<td>add2num/ISeriesImport.wSDL</td>
<td>Contains the WSDL file configured for the resource adapter</td>
</tr>
</tbody>
</table>
2) ADD2NUM_SAVF_PCML.zip – Zip archive includes source of sample RPG program and generated PCML file

<table>
<thead>
<tr>
<th>File name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADD2NUMSAV.SAVF</td>
<td>SAVF file which has the source code for RPG sample program. Please refer your i5/OS documentation for how to import this into IBM i system. Alternatively you can manually create a simple RPGLE to add two numbers like the one given below. The RPGLE source view is captured with the help of 'IBM Rational Developer for System i' tool.</td>
</tr>
<tr>
<td>ADD2NUM.PCML</td>
<td>PCML for sample RPG</td>
</tr>
</tbody>
</table>

Figure 1 - Sample RPG program as shown in ‘IBM Rational Developer for System i’ tool’s remote editor
Chapter 3. Outbound Processing – Running the RPG program

This tutorial demonstrates how WebSphere Adapter for IBM i V6.2 can be used to run an RPG program on an IBM i system.

Configuring the adapter for outbound processing

Run the external service wizard to specify business objects, services, and the configuration to be used in this tutorial

Creating the project

1. Launch WebSphere Integration Developer by clicking Start > Programs > IBM WebSphere > Integration Developer 6.2.

2. In WebSphere Integration Developer, switch to the Business Integration perspective by clicking Window > Open perspective > Other. In the Select perspective window, select Show all, then select Business Integration from the list and click OK.

3. Create a new module by clicking File > New > Module.

4. In the Module Name field, type add2num, and click Finish.
5. If it is not already open, open the Assembly Diagram of the module just created, by double clicking it.
6. From the Outbound Adapters section of the Palette section, drag and drop iSeries onto the Assembly Diagram editor. A window opens similar to one below.

7. Click IBM WebSphere Adapter for IBM i (IBM : 6.2.0) and click Next.

8. Select an appropriate run time from the Target runtime list. (WebSphere Process Server 6.2 is used as the run time in this tutorial.) Click Next.
Setting connection properties for the external service wizard

1. In the Discovery Configuration window, enter the IBM i server connection information such as the host name, user name, password, and path to the folder on the IBM i system for object discovery. From the **Object type to discover** list, select **Report Program Generation**.
2. After you have entered all properties, click **Next**.

**Generating business object definitions and related artifacts**

1. In the Object Discovery and Selection window, in the Discovered objects pane, all objects are displayed
2. In the Discovered Objects pane, select the PCML you want to work with, click the arrow (>) button and click **Next**.
3. In the Configure Composite Properties window, the operation CallPGM is listed in the Operations for selected business objects pane. Because this is the only operation supported for calling RPG programs, you cannot use the Add and Remove options. Click Next.
4. Clear the Specify a Java Authentication and Authorization Services (JAAS) alias security credential check box. The Host name and user name fields will already be populated. In the Password field, type the password and click Next.
5. A default name is provided for the interface. Click **Finish** to complete the configuration.
Deploying the module to the test environment

1. The result of running the external service wizard is an SCA module that contains an EIS import or export. Install this SCA module in the WebSphere Integration Developer integration test client. If WebSphere Process Server is not in “Started” state, start the server.

2. Add the module you created earlier to the server using the server panel in WebSphere Integration Developer. Right-click the server, and then select Add and remove projects.
3. Add the SCA module to the server by moving them from left to right pane with the click of ‘Add >’ button.

4. Click **Finish**.
Ensure that the Application started message is displayed in your console, as shown in the following figure.
Testing the assembled adapter application

1. Test the assembled adapter application by using the WebSphere Integration Developer integration test client.

2. Right-click the ‘ISeriesImport’ component, and select ‘Test Component’.
3. In the new editor window that opens, select `callpgmADD2NUM` from the **Operation** list, as shown in the following figure.
4. Type values for the input business objects and specify the required parameters to carry out the CallPGM operation on the RPG program. For example you can specify 2 and 4 as values for GET1 & GET2 respectively as shown below.

6. Run the service by clicking the continue button and select the **Use this as the default and do not ask again** check box and click **Finish**.
The value returned from the underlying RPG is populated in the same business object (GET3 attribute returns the value of GET1+GET2), as shown in the following figure.

Figure 18

Figure 19
Chapter 4. Introduction to Outbound/Inbound for Data Queues

The WebSphere® Adapter for IBM® i V6.2 exchanges business data between system i and J2EE applications. The adapter retrieves from and writes to the data queue and runs the RPG programs.

The document demonstrates two scenarios:

1. PutQueue and GetQueue outbound operations for data queues
2. Inbound operations for data queues

These scenarios demonstrate how WebSphere Adapter for IBM i V6.2 performs inbound and outbound operations. Everything you need to complete for each tutorial is contained in the tutorial. If you have performed the prerequisite tasks, you can complete each tutorial in under an hour.

Learning objectives

After completing a tutorial, you should be able to perform the following tasks:

- Create an adapter project in WebSphere Integration Developer.
- Discover services and associated business objects from the enterprise information system (EIS) and make them part of the adapter project.
- Create a deployable module that you install on WebSphere Process Server or WebSphere Enterprise Service Bus.
- Test the module and validate the results.

Audience

These tutorials are for integration developers who design, assemble, test, and deploy business integration solutions.
Software prerequisites

To use these tutorials, you must have the following applications installed:

- WebSphere Integration Developer version 6.2
- WebSphere Process Server version 6.2
Chapter 5. Preparing to run through the tutorial

Configuration prerequisites

Before starting this tutorial, create a data queue on an IBM i system.

Extracting the sample files

The artifacts created using the external service wizard will contain the files which are similar to the listed in the below table.

Sample Artifacts here is DTAQOutbound.zip.

<table>
<thead>
<tr>
<th>Contents of DTAQOutbound.zip</th>
</tr>
</thead>
<tbody>
<tr>
<td>File name</td>
</tr>
<tr>
<td>Description</td>
</tr>
<tr>
<td>DTAQOutbound/ISeriesOutboundInterface.import</td>
</tr>
<tr>
<td>DTAQOutbound/Fifoq.xsd</td>
</tr>
<tr>
<td>DTAQOutbound/EmptyGetQueueBO.xsd</td>
</tr>
<tr>
<td>DTAQOutbound/ISeriesOutboundInterface.wsdl</td>
</tr>
</tbody>
</table>
Sample Artifacts here is DTAQInbound.zip

<table>
<thead>
<tr>
<th>Contents of DTAQInbound.zip</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>File name</td>
<td>Description</td>
</tr>
<tr>
<td>DTAQInbound/ISeriesInboundInterface.export</td>
<td>Contains the SCA export for the resource adapter.</td>
</tr>
<tr>
<td>DTAQInbound/Fifoq.xsd</td>
<td>Business object definition for the Customer business function.</td>
</tr>
<tr>
<td>DTAQInbound/ISeriesInboundInterface.wsdl</td>
<td>Contains the WSDL file configured for the resource adapter</td>
</tr>
</tbody>
</table>
Chapter 6. Outbound Processing – Writing the message to (PutQueue) and reading the message from (GetQueue) the data queue.

This tutorial demonstrates how you can use WebSphere Adapter for IBM i V6.2 to put and get a string message from the data queue.

Configuring the adapter for outbound processing

Run the external service wizard to specify business objects, services, and the configuration to be used in this tutorial

Creating the project

1. Launch WebSphere Integration Developer by clicking Start > Programs > IBM WebSphere > Integration Developer 6.2.

2. Switch to the Business Integration perspective by clicking Window > Open perspective > Other. In the Select perspective window, select Show all, then select Business Integration from the list and click OK.

3. Create a new module by clicking File > New > Module.

4. In the Module Name field, type DTAQOutbound. Click Next.
6. Accept the default settings and click **Finish**.
After the module is created, the folder structure will look like that shown in the following figure.

Figure 3

9. Launch the external service wizard by right-clicking the module (DTAQOutbound) and selecting New > external service.
11. In the External Service window, click the + sign for the adapter to expand it and select iSeries. Click **Next**.
Figure 5

12. Select **IBM WebSphere Adapter for IBM i** (IBM : 6.2.0.0) and click **Next**.
Figure 6
13. Ensure that the ios RAR file is selected in the **Connector project** field, and in the **Target runtime** field select **WebSphere Process Server v6.2**. Click **Next**.

![Adapter Import](image)

**Adapter Import**
Import a resource adapter archive (RAR) from the file system to create a connector project for the adapter.

- **Archive file:** C:\IBM\wix620\ResourceAdapters\Ios_6.2.0.0\deploy\ios.rar
- **Connector project:** ios
- **Target runtime:** WebSphere Process Server v6.2

![Figure 7](image)

14. In the Processing Direction window, select **Outbound**. Click **Next**.
Figure 8

**Setting connection properties for the external service wizard**

1. Type the IBM i server connection information such as the host name, user name, password, and path to the folder on the IBM i system for object discovery. From the **Object type to discover** list select **Data Queue**.
2. After you have entered all properties, click **Next**.

**Generating business object definitions and related artifacts**

In the Object Discovery and Selection window, in the Discovered objects pane, all data queues are displayed.
Object Discovery and Selection

Expand nodes to view and select discovered objects. Use filtering, if necessary, to limit number of objects displayed.

Figure 10

1. Select the data queue from the Discovered objects pane and move it to the Selected objects pane. Click Next.
2. In the Configure Composite Properties window, the operations PutQueue and GetQueue are listed in the **Operations for selected business objects** pane. Click on Next button.
3. Clear the **Specify a Java Authentication and Authorization Services (JAAS) alias security credential** check box. The **Host name** and **user name** fields will already be populated. Type the password in the **Password** field and click **Next**.
4. A default name is provided for the interface. Click **Finish** to complete the configuration.
6. Verify the results by checking the artifacts generated in business integration view for the module as shown in below screen.
Deploying the module to the test environment

The result of running the external service wizard is an SCA module that contains an enterprise information system (EIS) import or export. Install this SCA module in the WebSphere Integration Developer integration test client.

1. Start WebSphere Process server.

2. Add the module you created earlier to the server by using the server panel in WebSphere Integration Developer. Right-click the server, and then select Add and remove projects.
3. Select the DTAQOutbound module and click on Add button, then click **Finish**.
4. In the Adding/Removing Projects window, click **OK**.

![Adding/Removing Projects](image)

Figure 18

---

**Testing the assembled adapter application**

1. Test the assembled adapter application by using the WebSphere Integration Developer integration test client.

2. Open the test component by right-clicking **ISeriesOutboundInterface** then selecting **Test Component**.
3. Select putqueueFifoq from the operation as from the operation.
4. Go to the Initial request parameters and enter the text ‘test message’ for messageContent request field.
Figure 21

6. Run the service by clicking the continue button and selecting the **Use this as the default and do not ask again** check box. Click **Finish**.
7. If security is set up, the User Login – Default Module Test window opens. Click **OK**.

8. The result will be displayed as shown in the below screen.
9. To test the GetQueue operation on the data queue, click the button. This opens another test component. In the **Operation** field, elect `getqueue_fifoq` and click .
Figure 25

The message from the data queue is displayed as return message.
Figure 26
Chapter 7. Inbound processing on a data queue

This tutorial demonstrates how you can use WebSphere Adapter for IBM i V6.2.0.0 to poll the data queue and send the messages to the endpoint application.

Configuring the adapter for inbound processing

Run the external service wizard to specify business objects, services, and the configuration to be used in this tutorial.

Creating the project

1. Launch WebSphere Integration Developer by clicking Start > Programs > IBM WebSphere Integration Developer 6.2.

2. In WebSphere Integration Developer, switch to the Business Integration perspective by clicking Window > Open perspective > Other. In the Select perspective window, select Show all, then select Business Integration from the list and click OK.

3. Create a new module by clicking File > New > Module.

4. In the Module Name field, type DTAQInbound and click Next.
5. In the Solution Creation window, click **Finish**.
6. After the module is created, the folder structure will look like that shown in the following figure.

7. Launch the external service wizard by right-clicking the DTAQInbound module and selecting **New > External Service.**
8. In the External Service window, click the + sign for the adapter to expand it and select **iSeries**. Click **Next**.

9. Select **iSeries** and click **Next**.
External Service

Select the type of service to create.

Filter: type filter text

Available Types:
- Adapters
- JAVA
- Messaging
- Registries

Description:

Finish  Cancel
9. Click **IBM WebSphere Adapter for IBM i** and click **Next**.
10. Ensure that the ios RAR file is selected in the **Connector Project** field, and from the **Target runtime** list, select **WebSphere Process Server v6.2**. Click **Next**.
11. Select the **Inbound** check box and click **Next**.
Setting connection properties for the external service wizard

1. Enter the IBM i server connection information such as the host name, user name, password and path to the folder on the IBM i for object discovery. The Object type to discover selection filed will be disabled.
Figure 34

2. After all properties are entered, click Next.
Generating business object definitions and related artifacts

On the metadata tree panel, all data queues will be displayed.

Figure 35

2. Select the data queue from the **Discovered objects pane**, and move it to the **Selected objects pane**. Click **Next**.
3. In the Configure Composite Properties window, the Emit operation is displayed in the Operations for selected business objects pane. Click Next.
4. Clear the option **Specify a Java Authentication and Authorization Services (JAAS) alias security credential** check box. The **Host name, user name** and **Poll Queue Path** fields will already be populated. The Poll Queue Path field is disabled. Type the password, payload staging queue path and control language program path in the appropriate fields, and click Next.
Figure 38
Figure 39
5. A default name is provided for the interface. Click **Finish** to complete the configuration.
6. Verify the results by checking the artifacts generated in business integration view for the module as shown in below screen.
7. Drag and drop the Java Component from the Palette and draw a wire from 
<code>ISeriesInboundInterface</code> to Component1.

8. Implement the java component with the logic for processing the event. This can be done by double clicking the Component1.

The below screen will be opened after double clicking the component1.
Deploying the module to the test environment

The result of running the external service wizard is an SCA module that contains an enterprise information system (EIS) import or export. Install this SCA module in the WebSphere Integration Developer integration test client.


2. Add the module you created earlier to the server using the server panel in WebSphere Integration Developer. Right-click the server, and then select Add and Remove Projects.
2. Select the DTAQInbound module and click on add button, then click **Finish**
Figure 45
Testing the assembled adapter application

Test the assembled adapter application using the WebSphere Integration Developer integration test client.

1. Right click the adapter module, **DTAQInbound** then select **Test > Attach**.
Figure 47

This will open the following screen.

Figure 48
2. Run the service by clicking the continue button which will deliver the event to the endpoint.

3. Verify this by checking for the endpoint messages in System.Out file of WebSphere Process Server or by viewing the server console output in WebSphere Integration Developer, as shown in the following screen or WebSphere Process Server logs.

Figure 49
WebSphere® Adapter for IBM® i V6.2 exchanges business data between system i and J2EE applications. The adapter retrieves data from and writes to the data queue and runs RPG programs.

The document demonstrates two scenarios:

1. PutQueue and GetQueue outbound operations for keyed data queues

2. Inbound operations for keyed data queues

These tutorials demonstrate how WebSphere Adapter for IBM i V6.2 performs inbound and outbound operations. To gain practical knowledge in setting up and deploying the adapter, complete one or more of the tutorials. Everything you need to complete each tutorial is contained in the tutorial. If you have performed the prerequisite tasks, you can complete each tutorial in less than an hour.

Learning objectives

After completing a tutorial, you should be able to perform the following tasks:

Create an adapter project in WebSphere Integration Developer.

Discover services and associated business objects from the enterprise information system (EIS) and make them part of the adapter project.

Create a deployable module that you install on WebSphere Process Server or WebSphere Enterprise Service Bus.

Test the module and validate the results.

Audience

These tutorials are for integration developers who design, assemble, test, and deploy business integration solutions.
Software Prerequisites

To use these tutorials, you must have the following applications installed:

WebSphere Integration Developer version 6.2

WebSphere Process Server version 6.2
Chapter 9. Preparing to run through the tutorial

Configuration prerequisites

Before doing any tutorial testing, complete the following task:

1. Create a keyed data queue on an IBM i system.

Extract the sample files

Replicas of the artifacts that you create when using the external service wizard are provided as sample files for your reference. Use these files to verify that the files you create with the external service wizard are correct.

Note that the values for the i5/OS host name, user name, and password in the sample artifacts are from the IBM test lab. You need to change or set them appropriately for your environment.

Following table lists the artifacts that are shipped as part of samples which user can use to verify when handling Keyed Data Queue.

1) DTAQOutbound.zip - Project Interchange file which includes generated sample artifacts for handling messages on Keyed Data Queues.

<table>
<thead>
<tr>
<th>File name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>KDTAQOutbound</td>
<td>Contains the SCA import for the resource adapter.</td>
</tr>
<tr>
<td>KDTAQOutbound/ISeriesOutboundInterface.impor t</td>
<td></td>
</tr>
<tr>
<td>KDTAQOutbound/Keyedq.xsd</td>
<td>Business object definition for the Customer business function.</td>
</tr>
<tr>
<td>KDTAQOutbound/EmptyGetQueueBO.xsd</td>
<td>Business object definition for the business object container.</td>
</tr>
<tr>
<td>KDTAQOutbound/ISeriesOutboundInterface.wsdl</td>
<td>Contains the WSDL file configured for</td>
</tr>
</tbody>
</table>
2) KDTAQInbound.zip - Project Interchange file which includes generated sample artifacts for polling messages on a Keyed Data Queue.

<table>
<thead>
<tr>
<th>File name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>KDTAQInbound</td>
<td></td>
</tr>
<tr>
<td>KDTAQInbound/ISeriesInboundInterface.exp</td>
<td>Contains the SCA export for the resource adapter.</td>
</tr>
<tr>
<td>KDTAQInbound/Keyedtaqkeyedq.xsd</td>
<td>Business object definition for the Customer business function.</td>
</tr>
<tr>
<td>KDTAQInbound/ISeriesInboundInterface.wsdl</td>
<td>Contains the WSDL file configured for the resource adapter</td>
</tr>
</tbody>
</table>
Chapter 10. Outbound Processing – Writing (PutQueue) the message to and reading (GetQueue) the message from the keyed data queue

This tutorial demonstrates how WebSphere Adapter for IBM i V6.2 can be used to put and get a string message from the keyed data queue.

Configuring the adapter for outbound processing

Run the external service wizard to specify business objects, services, and the configuration to be used in this tutorial

Creating the project

1. Launch WebSphere Integration Developer by selecting Start > Programs > IBM WebSphere > Integration Developer 6.2.

2. In WebSphere Integration Developer, switch to the Business Integration perspective. To do this, click Window > Open perspective > Other. In the Select perspective window, select Show all, then select Business Integration from the list and click OK.

3. Create a new module by selecting File > New > Module.

4. Type KDTAQOutbound in the Module Name field, and click Finish.
5. Launch the external service wizard by selecting File > New > Other > Business Integration > External Service.

6. Ensure that **Adapters** is selected and click **Next**.
7. Click IBM WebSphere Adapter for IBM i and click Next.
8. Ensure that `ios.rar` is selected and click **Next**.
9. Select **Outbound** and click **Next**
Setting connection properties for the external service wizard

1. In the Discovery Configuration window, enter the IBM i server connection information such as the host name, user name, password, path to the folder on IBM i for object discovery and the object type to discover to keyed data queue.
2. After all properties are entered, click **Next**.

**Generating business object definitions and related artifacts**

1. On the **Object Discovery and Selection** panel, all data queues will be displayed for the path to selected folder location.
2. Select the keyed data queue from the left Discovered object pane and click **Next** to move it to the Selected objects pane.
3. In the Configure Composite Properties window, the operations PutQueue and GetQueue are listed in the **Operations for selected business objects** pane. Select the operations for the data queue by clicking the **Add** button. Click **Next**.
4. Clear the Specify a Java Authentication and Authorization Services (JAAS) alias security credential check box. The Host name and user name fields will be populated. Type the password in the Password field and click Next.
5. A default name is provided for the interface. Click **Finish** to complete the configuration.
6. The generated module components look as shown below.
Deploying the module to the test environment

The result of running the external service wizard is an SCA module that contains an EIS import or export. Now you will install this SCA module in the WebSphere Integration Developer integration test client.

1. Add the module you created earlier to the server by using the server panel in WebSphere Integration Developer. Right-click the server, and then select Add and Remove Projects.

2. Add the SCA module to the server by selecting it and clicking Add. Then click Finish.
Figure 14
Figure 15
Testing the assembled adapter application

1. Test the assembled adapter application using the WebSphere Integration Developer integration test client.

2. Select the service you want, and click Test > Test Module.

3. Select the operation as putqueueKeyedq from the operation list.

4. Populate values for input business objects, by selecting the operation putQueueKeyedq.

5. Specify the required parameters (messageKey, messageContent) to perform the PutQueue operation on the Keyedq data queue.

Figure 16
6. Run the service by clicking the continue icon ( ). Select the **Use this as the default and do not ask again** check box and click **Finish**.
Figure 18

The output looks as shown below.
7. To test the GetQueue operation on data queue click the button. Another test component is opened. Select the `putqueueKeyedq` from the Operation list and click .

The message from the data queue is displayed as return message, as shown in the following figure.
Figure 21
Chapter 11. Inbound processing on the keyed data queue

This tutorial demonstrates how WebSphere Adapter for IBM i V6.2.0.0 can be used to poll the keyed data queue and send the messages to the endpoint application.

Configuring the adapter for inbound processing

Run the external service wizard to specify business objects, services, and the configuration to be used in this tutorial.

Creating the project

1. Launch WebSphere Integration Developer by clicking Start > Programs > IBM WebSphere > Integration Developer 6.2.

2. In WebSphere Integration Developer, switch to the Business Integration perspective by clicking Window > Open perspective > Other. In the Select perspective window, select Show all, then select Business Integration from the list and click OK.

3. Create a new module by clicking File > New > Module.

4. Type KDTAQInbound in the Module Name field and click Finish.
5. Launch the external service wizard by selecting File > New > Other > Business Integration > External Service.

6. Ensure that Adapters is selected, select iSeries and click Next.
Figure 24

7. Click IBM WebSphere Adapter for IBM i (IBM : 6.2) and click Next
Figure 25

8. Ensure that the ios RAR file is selected and click **Next**.
9. Select **Inbound** and click **Next**.
Figure 27

**Setting connection properties for the external service wizard**

1. Enter the IBM i server connection information such as the host name, user name, password and path to the folder on the IBM i system for object discovery. The **Object type to discover** field will be disabled.
2. After you have entered all properties, click **Next**.

**Generating business object definitions and related artifacts**

1. In the Discovered objects pane, all data queues are be displayed so that you can select the path to the folder.
2. Select the keyed data queue from the *Discovered objects* pane and click the > button to move it to the *Selected objects* pane. Click Next.
3. In the Configure Composite Properties window, the Emit operation is displayed and is disabled. Click Next.
4. Clear the Specify a Java Authentication and Authorization Services (JAAS) alias security credential. The Host name, user name, and Poll Queue Path fields will be populated in this window. The poll queue path field, however, is disabled. Type the password, in the Payload Staging Queue Path field and Polling Control Language Program path fields and click Next.
The advanced properties looks as shown below.
5. A default name is provided for the interface. Click **Finish** to complete the configuration.

Figure 32

6. The generated module contains the following components in it.

Figure 33
7. Generate a Java™ component and draw a wire from `ISeriesInboundInterface` to `Component1`.

8. Generate implementation for the java component interface as shown below. You can choose “default package” when asked for package for the implementation.
Figure 36

Figure 37
The generated Java implementation is shown in the following screen capture.

![Java code screenshot](image)

Figure 38

---

**Deploying the module to the test environment**

The result of running the external service wizard is an SCA module that contains an EIS import or export. Install this SCA module in the WebSphere Integration Developer integration test client.


2. Add the module you created earlier to the server by using the server panel in WebSphere Integration Developer. Right-click the server, and then select **Add and Remove Projects**.
Figure 39

2. Add the SCA module from the left frame to the right frame by selecting it and clicking **Add**. Then click **Finish**.
Figure 40
Testing the assembled adapter application

Test the assembled adapter application by using the WebSphere Integration Developer integration test client.

1. Right-click the adapter module, **DTAQInbound**, then select **Test > Attach**.

The relevant business object is delivered to the endpoint

2. Verify that the business object has been delivered by either checking for the endpoint messages in the System.Out file of WebSphere Process Server or by viewing the server console output in WebSphere Integration Developer.
Figure 42

This will open the following screen.
2. Run the service by clicking the continue button which will deliver the event to the endpoint.

3. Verify this by checking for the endpoint messages in System.Out file of WebSphere Process Server or by viewing the server console output in WebSphere Integration Developer, as shown in the following screen or WebSphere Process Server logs.
Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:
INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication.

IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites.

The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation Department
2Z4A/SOM1 294 Route 100
Somers, NY 10589-0100 U.S.A.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.
This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include

the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming interface information

Programming interface information, if provided, is intended to help you create application software using this program.

General-use programming interfaces allow you to write application software that obtain the services of this program's tools.

However, this information may also contain diagnosis, modification, and tuning information. Diagnosis, modification and tuning information is provided to help you debug your application software.

Warning:

Do not use this diagnosis, modification, and tuning information as a programming interface because it is subject to change.
**Trademarks**

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A complete and current list of IBM trademarks is available on the Web at [http://www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml)

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.*

Other company, product, or service names may be trademarks or service marks of others.

This product includes software developed by the Eclipse Project ([http://www.eclipse.org](http://www.eclipse.org)).