

WebSphere® software



BPM 7.5 – Creating a Hello World iWidget

Ashok Iyengar
ashoki@us.ibm.com

Contents

Introduction.....	3
Environment.....	3
Section A – Creating a simple widget.....	4
Section B – Packaging the widget	17
Section C – Deploying the widget	20
Section D – Display the widget	22
Conclusion	23
References.....	23

Introduction

IBM Business Process Management (BPM) software stack is now at v7.5. This software is based on a service-oriented architecture (SOA) and embraces open standards.

IBM BPM v7.5 provides a Web 2.0 widget-based User Interface called Business Space that can display individual widgets or a mashup of widgets to create dashboards. It is browser-based UI that lets business users interact with content from products in the IBM Business Process Management and Connectivity portfolio. The business spaces you create are collections of related Web content that provide you with insight into your business and the capability to react to changes in it.

This document was put together from the snippets of instructions in the IBM BPM v7.5 infocenter. It shows how to create a simple Hello World Widget and then guides you through the steps of packaging it, registering it, and deploying it to the widget palette in Business Space v7.5.

Environment

Machine A (IID)

Software	Version	Installation Location	Variable
IBM Integration Designer	7.5	C:\IBM\IID75	<IID_HOME>
IBM Process Center	7.5	C:\IBM\WebSphere\Appserver	<PC_HOME>

Section A – Creating a simple widget

In IBM BPM v7.5 the tool of choice, for creating Eclipse projects, is the IBM Integration Designer (IID). As the product infocenter points out you can also use Rational Application Developer, WebSphere sMash, Portlet Factory, or any Java based tooling.

Step 1: Creating a new Eclipse project

You will create a new Eclipse project to contain your new widget. This step is required in order to create all the necessary artifacts that must be contained in your widget WAR file. To create the new project, do the following:

1. Make sure you are in IID's Business Integration perspective.
2. Click **File->New->Project**.
 - a. In the **New Project** window, choose **Web->iWidget Project**.
 - b. Click **Next**.

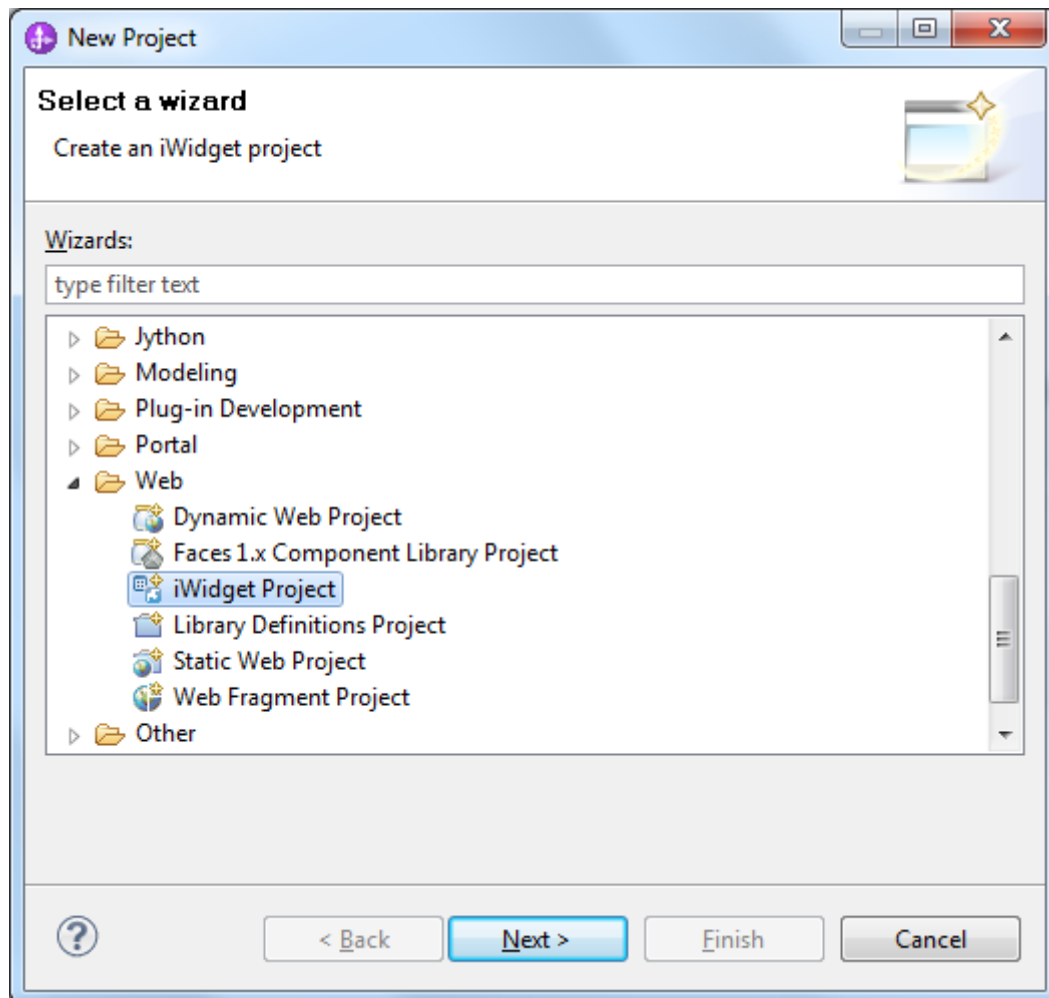


Figure A1

3. Choose **Web and JavaEE technologies** as the technology type.

Recommendation: Do not check the box to remember the workspace preference decision.

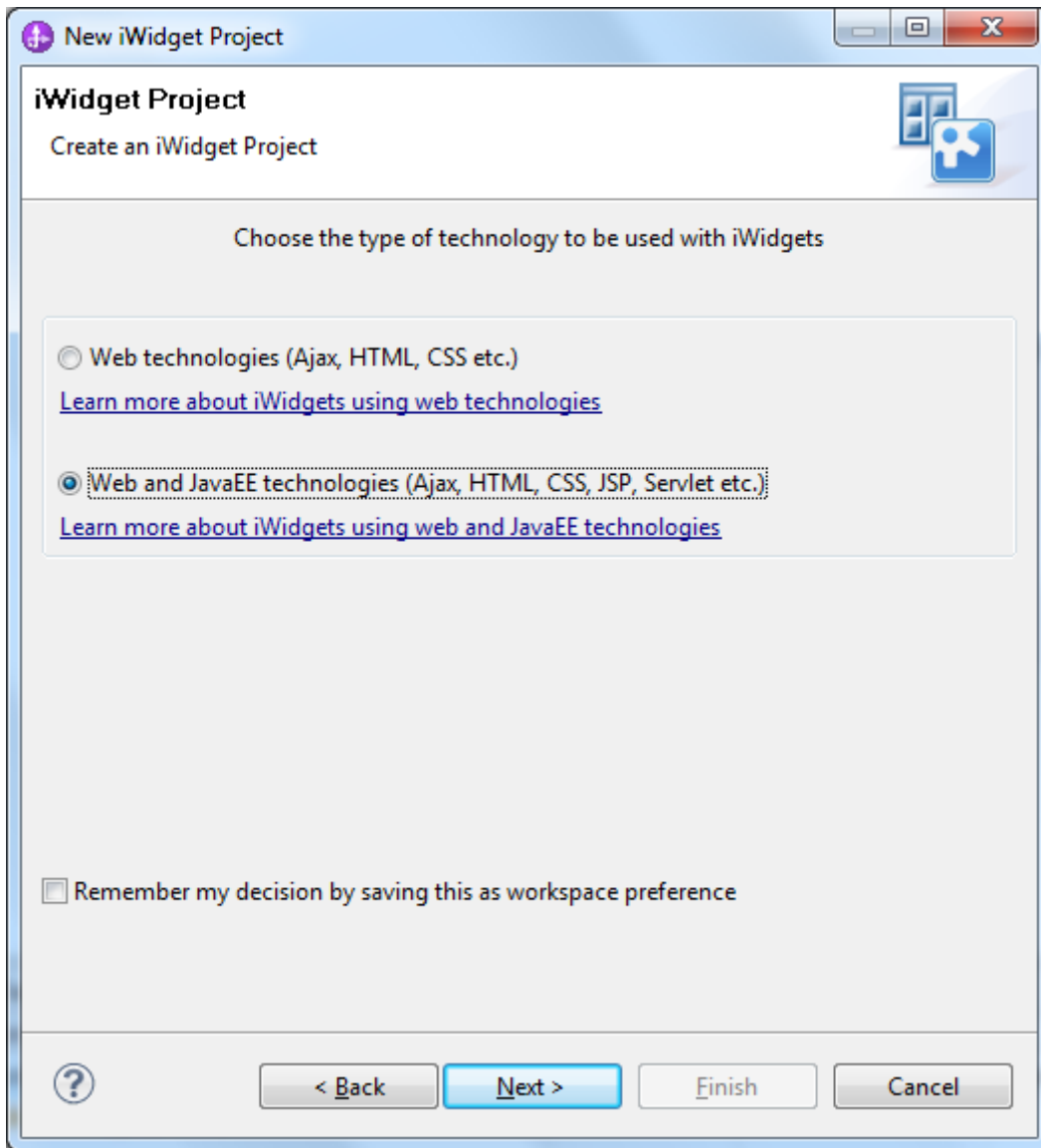


Figure A2

4. In the Dynamic Web Project window
 - a. Enter a Project name – *HelloWorldWidget*
 - b. Check the **default location**
 - c. Choose *IBM Process Server v7.5* for Target runtime
 - d. Check the box to *Add project to an EAR*

- e. Accept the default EAR file name or you may drop the suffix EAR to suit the widget naming conventions of WIDGET_NAME.ear
- f. Click **Finish**

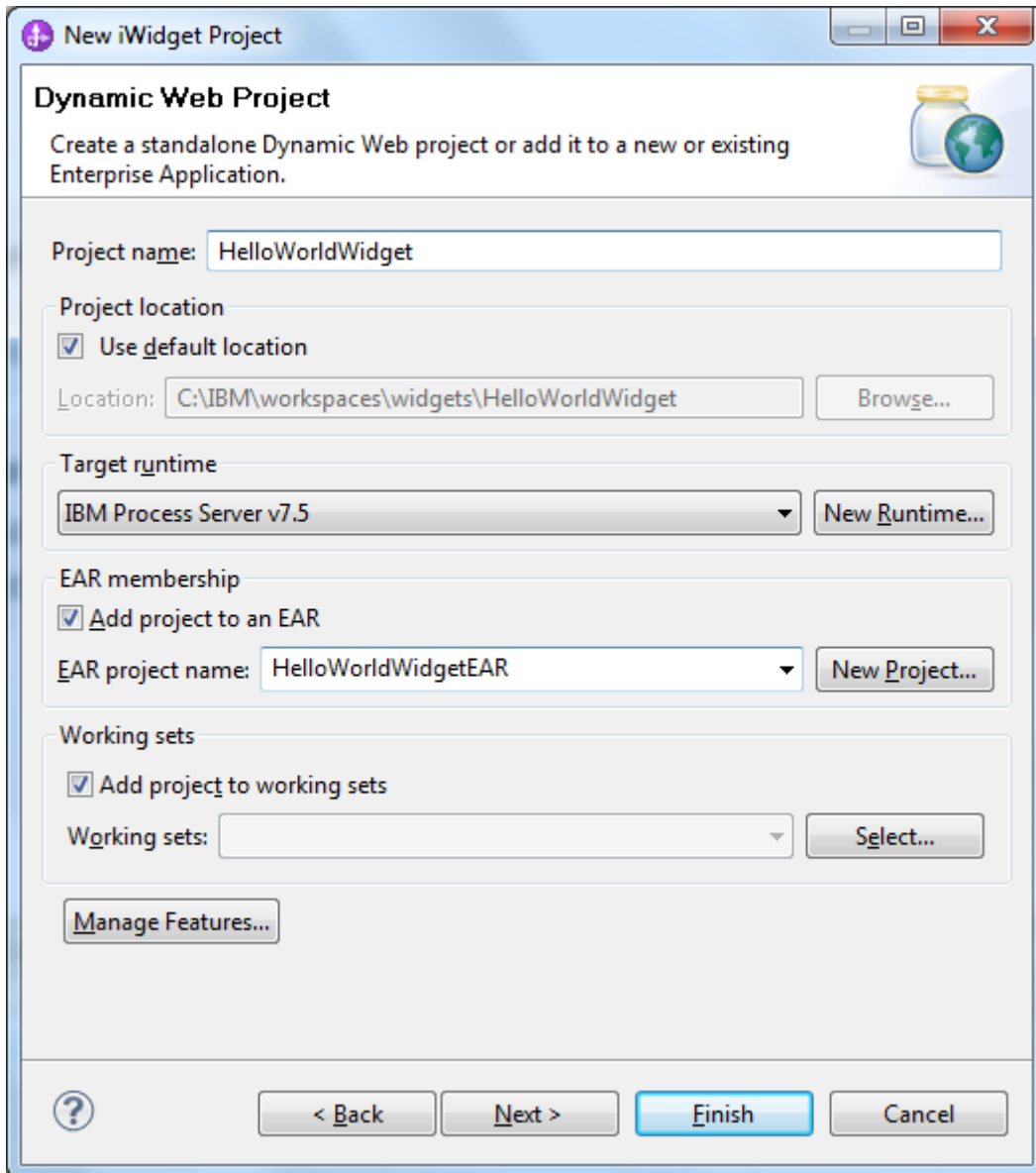


Figure A3

Note: If you click Manage Features button, you will see the project facets. Notice that by default, Dynamic Web Module version is 2.5 and iWidgets is version 1.1.

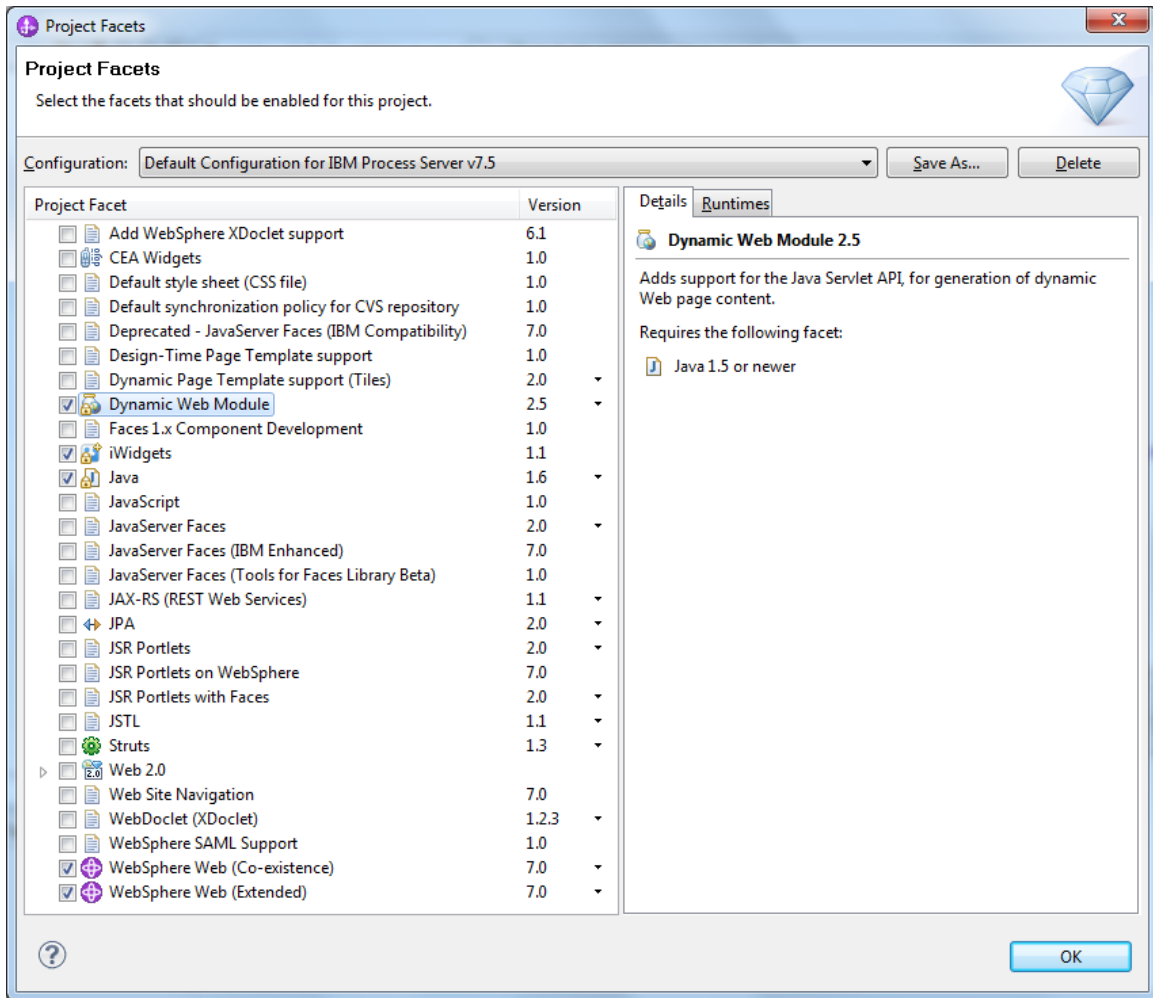


Figure A4

5. In the following screen you can choose from the popular widget types – Simple, Event Publisher, Event Subscriber, and Widget with JSP content. For this example, choose **Simple Widget**.
 - a. Do not check the Edit mode box because this widget will only display the text string ‘Hello World’.
 - b. Enter a Project name – *HelloWorldWidget*

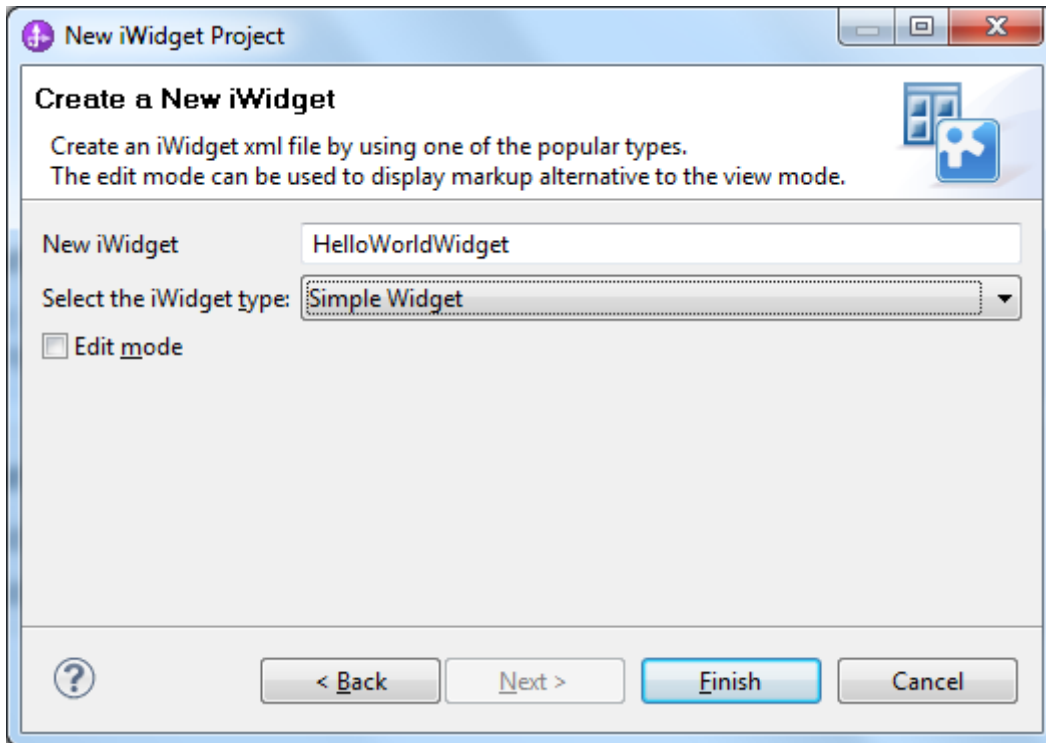


Figure A5

6. Click **Finish**. IID creates the new project, including all the required artifacts, in your Eclipse workspace.
 - a. You may choose not to switch perspectives.
7. In the **Enterprise Explorer** pane on the left, verify that your new ***HelloWorldWidget*** project structure looks similar to what is shown in Figure A6. Notice the corresponding ***HelloWorldWidgetEAR*** project.

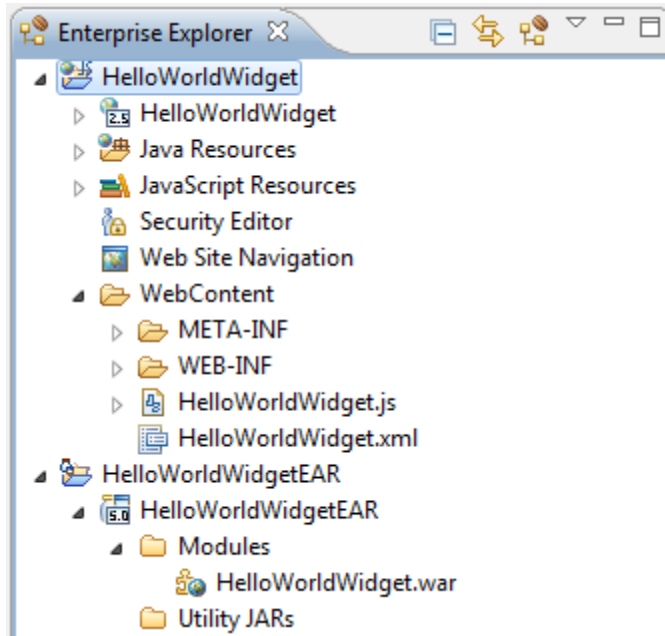


Figure A6

Note: When IID creates the iWidget project you will find a *mashup.properties* file under WebContent/Web-INF folder. It contains the contextRoot.

Step 2: Working with the widget XML file

You can use the default widget XML file or create a new one.

Tip: For Business Space, the naming convention for the widget definition file is *widgetName_iWidget.xml*.

8. If you notice the tooling generated a widget XML file – HelloWorldWidget.xml in the WebContent folder. You can use it as is and skip to step 9 or you can make a few changes and follow the naming convention as recommended in the BPM v7.5 product info center.
 - a. Rename HelloWorldWidget.xml to HelloWorldWidget_iWidget.xml by clicking the right-mouse button on the original file and choosing **Rename**.
 - i. Enter the new file name and click **OK**.
 - b. Under the WebContent folder create a new folder structure named iWidget/widgets/<WIDGET_NAME>. In this case it would be iWidget/widgets/HelloWorldWidget.
 - c. Move HelloWorldWidget_iWidget.xml to the HelloWorldWidget folder by clicking the right-mouse button on the file and choosing **Move**.
 - i. Choose the destination folder as shown in Figure A7.

ii. Click **OK**.

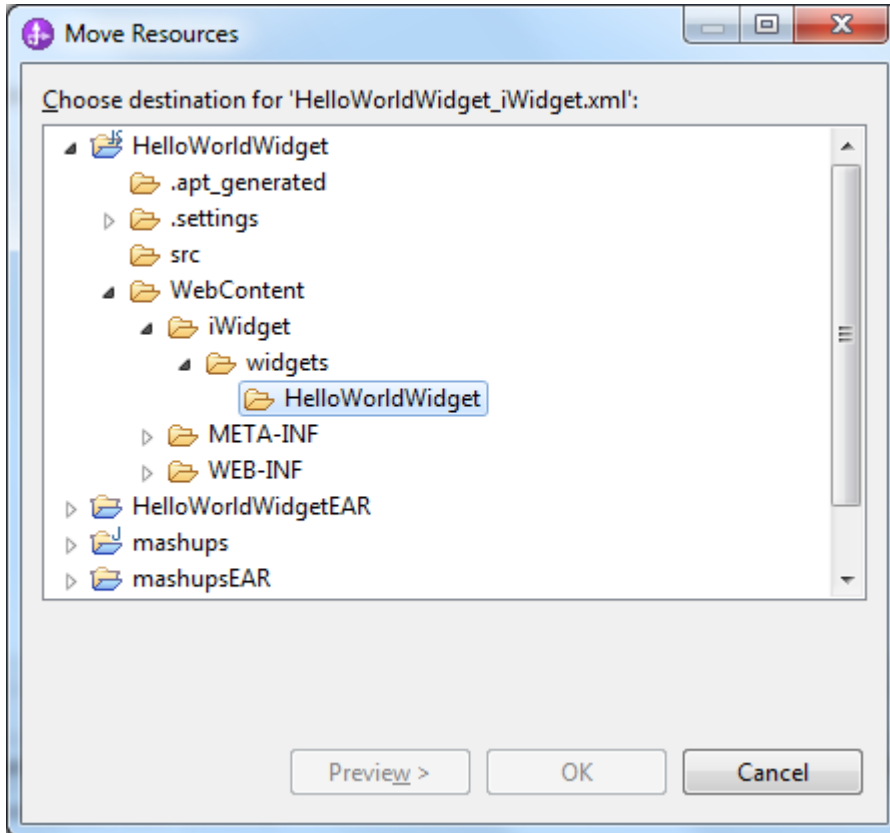
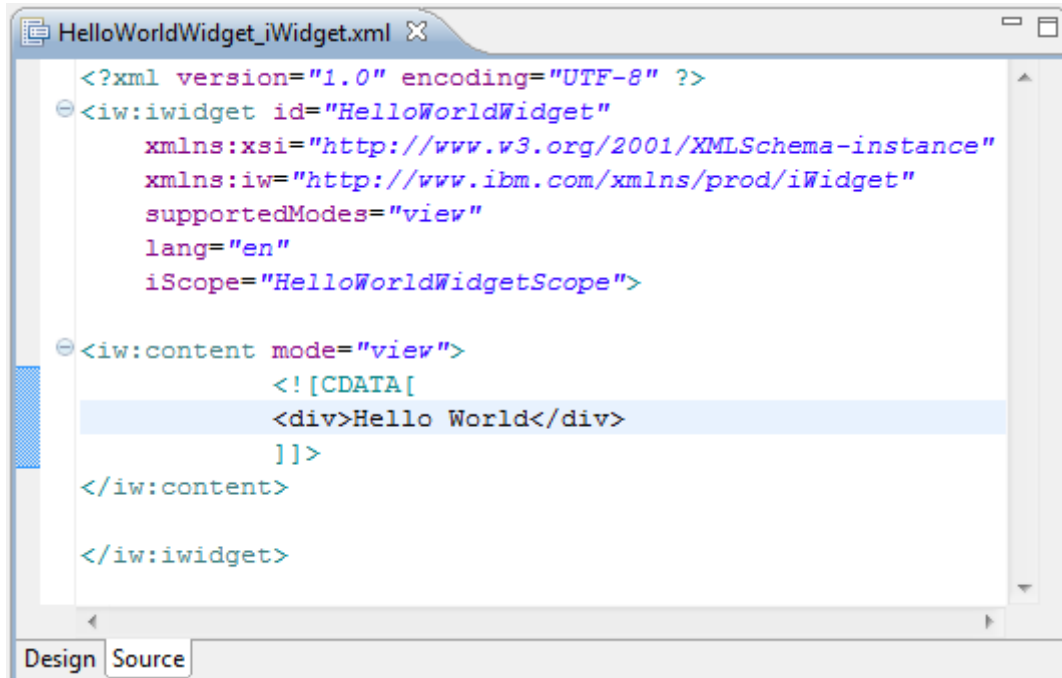


Figure A7

9. If you double-click the `HelloWorldWidget_iWidget.xml` file it will be opened in the XML Editor. Go to the Source tab and view the contents. You will notice:
 - a. The widget id is “HelloWorldWidget”
 - b. The supportedMode is ”view”
 - c. The generated iScope is “HelloWorldWidgetScope”
 - d. And the view mode content is “Hello World”

10. You may change the contents of the widget but for this example we will accept the default content. Close the *HelloWorldWidget_iWidget.xml* file.



```
<?xml version="1.0" encoding="UTF-8" ?>
<iw:iwidget id="HelloWorldWidget"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:iw="http://www.ibm.com/xmlns/prod/iWidget"
  supportedModes="view"
  lang="en"
  iScope="HelloWorldWidgetScope">
  <iw:content mode="view">
    <![CDATA[
      <div>Hello World</div>
    ]]>
  </iw:content>
</iw:iwidget>
```

Figure A8

Step 3: Adding the image file

It is a good idea to associate the widget with an image so that it can be identified in the widget palette alongside its name.

11. In the **Enterprise Explorer** expand WebContent->iWidget->widgets-HelloWorldWidget
 - a. Right-click and select **Import**
 - b. Go to **General->File System** and select **Next**
 - c. Browse for the image file like the generic_iWidget.gif shown in Figure A9.
 - d. Click **Finish**.



Figure A9

Step 4: Creating the catalog definition file

One of the most important files in the widget package is the catalog definitions file, which is another XML file that defines the widgets in the EAR file.

Note: When IID creates the iWidget project you will find a *catalog.xml* under WebContent/Web-INF folder. You may use that if you so choose.

12. Highlight the **HelloWorldWidget** project, right-mouse click and choose **New->Other->XML->XML File**
 - a. Click **Next**

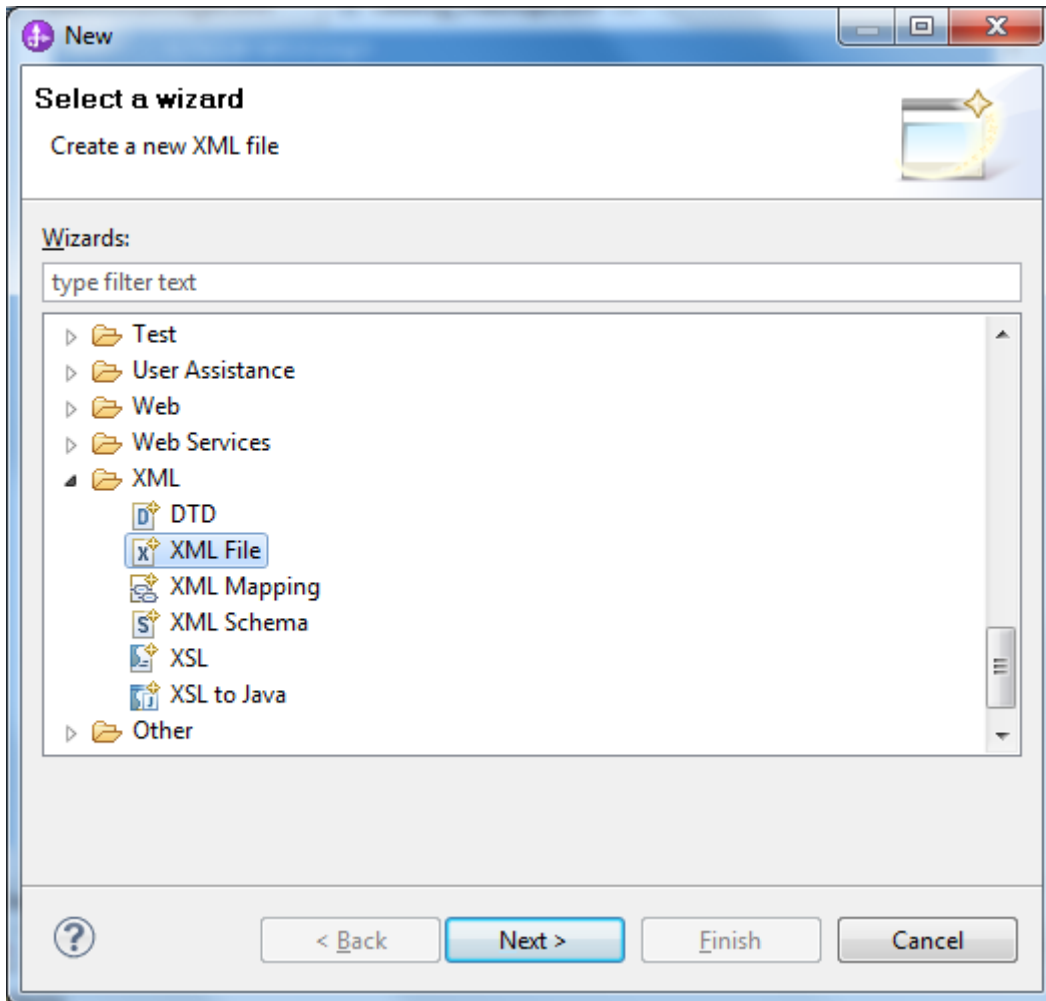


Figure A10

Tip: For Business Space, the naming convention for the catalog definition file is `catalog_catalogName.xml`. Do not put the catalog definitions file under WebContent folder. Keep it as a separate individual file or in a catalog folder.

- b. Make sure *HelloWorldWidget* is highlighted as the parent folder
- c. Name the new file *catalog_Hello.xml*
- d. Click **Finish**.

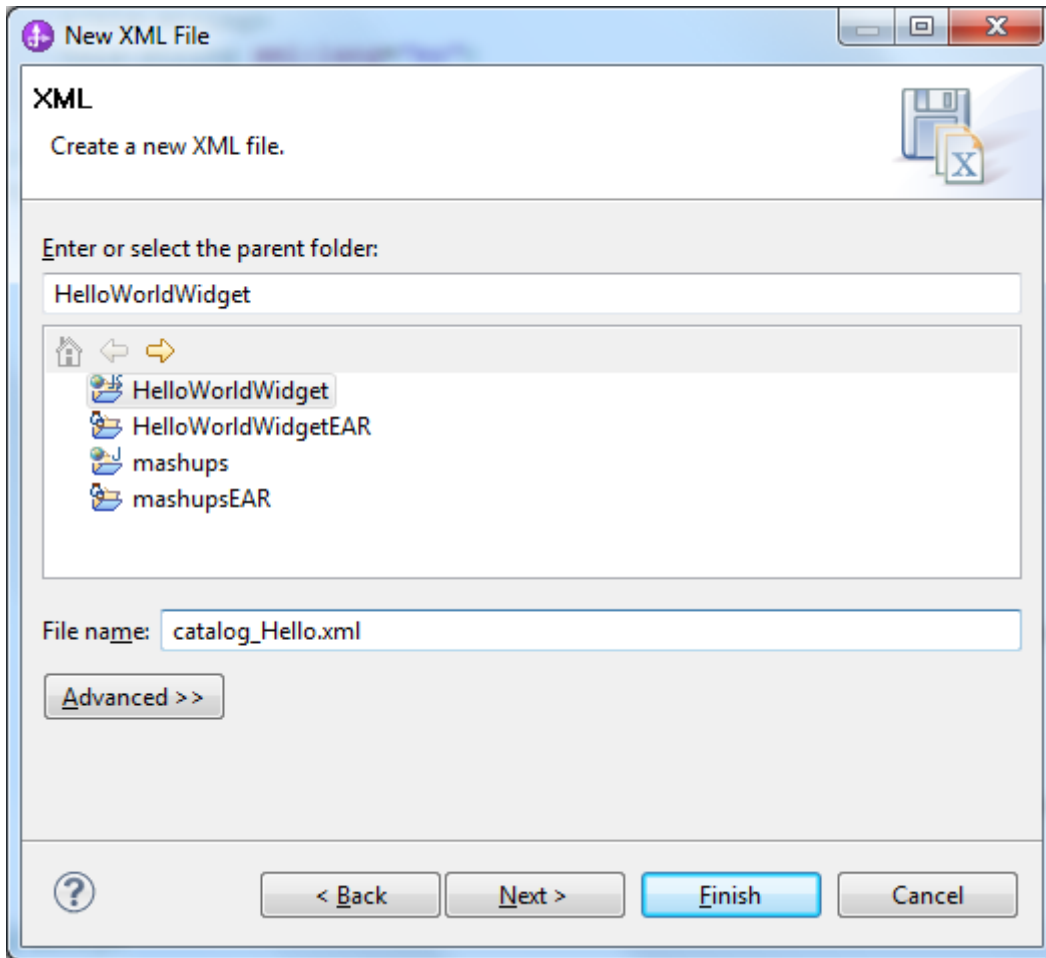


Figure A11

13. The *catalog_Hello.xml* file is open for editing.
 - a. Click the **Source** tab, and paste the following code into the window. If the xml version line is already generated, you do not need to duplicate it.
 - b. Notice the catalog id is *Hello* which is also part of the file name.
 - c. The *id* and *unique-name* are key values and should match the widget name.
 - d. And the defition URL, for iWidget.xml, should be accessible.

```
<?xml version="1.0" encoding="UTF-8"?>
<catalog id="Hello">
  <resource-type>Catalog</resource-type>
  <category name="Tools">
    <title>
      <nls-string xml:lang="en">Hello</nls-string>
      <nls-string xml:lang="es">Hola</nls-string>
    </title>
    <description>
      <nls-string xml:lang="en">
        Hello widgets
      </nls-string>
    </description>
  </category>
</catalog>
```

```

        </nls-string>
        <nls-string xml:lang="es">
            Hola widgets
        </nls-string>
    </description>
    <entry id="{com.ibm.bspace>HelloWorldWidget" unique-
name="{com.ibm.bspace>HelloWorldWidget">
        <title>
            <nls-string xml:lang="en">Hello World</nls-
string>
            <nls-string xml:lang="es">Hola el Mundo</nls-
string>
        </title>
        <description>
            <nls-string xml:lang="en">
                This widget displays the string Hello
World
            </nls-string>
            <nls-string lang="es">
                Este widget muestra la linea Hola Mundo
            </nls-string>
        </description>

        <definition>/HelloWorldWidget/iWidget/widgets/HelloWorldWidget/He
lloWorldWidget_iWidget.xml
        </definition>
        <content>http://www.ibm.com</content>
        <preview>http://www.ibm.com</preview>
        <icon>

        /HelloWorldWidget/iWidget/widgets/HelloWorldWidget/generic_iWidge
t.gif
        </icon>
        <metadata name="widgetParam" />
        <metadata name="onNewWire" />
    </entry>
</category>
</catalog>

```

Listing A1

14. Press **Ctrl + S** to save the file. Then close the file.

Step 5: Creating the endpoints file

Another file that goes in the widget package is the endpoints file, which is another XML file that defines the various widget endpoints. For our simple example, this step is optional.

Tip: For Business Space, the naming convention for the endpoints file is `endpoints_WidgetName.xml`. Again, do not put the `endpoints.xml` file under the `WebContent` folder. Keep it as a separate individual file.

15. Highlight the **HelloWorldWidget** project, right-mouse click and choose **New->Other->XML->XML File**
 - a. Click **Next**
 - b. Make sure *HelloWorldWidget* is highlighted as the parent folder
 - c. Name the new file *endpoints_HelloWorldWidget.xml*
 - d. Click **Finish**.

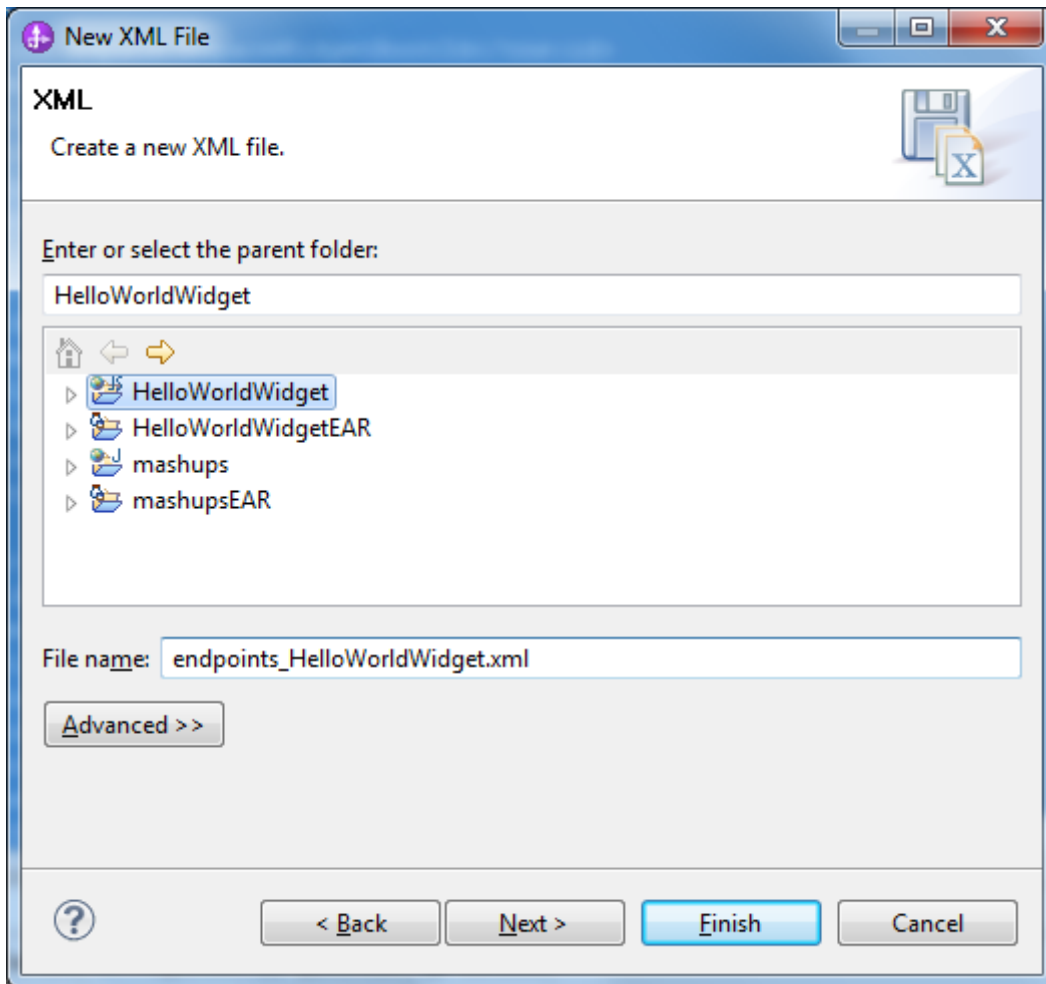


Figure A11

16. The *endpoints.xml* file is open for editing.
 - a. Click the **Source** tab, and paste the following code into the window. If the xml version line is already generated, you do not need to duplicate it.

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- START NON-TRANSLATABLE -->
<tns:BusinessSpaceRegistry
xmlns:tns="http://com.ibm.bspace/BusinessSpaceRegistry"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```

xsi:schemaLocation="http://com.ibm.bspace/BusinessSpaceRegistry
BusinessSpaceRegistry.xsd ">

<tns:Endpoint>
<tns:id>{com.ibm.bspace>HelloWorldWidgetId</tns:id>
<tns:type>{com.ibm.bspace>HelloWorldWidgetId</tns:type>
<tns:version>1.0.0.0</tns:version>
<tns:url>/HelloWorldWidget/</tns:url>
<tns:description>Location of Hello World widget</tns:description>
</tns:Endpoint>

<tns:Endpoint>
<tns:id>{com.ibm.bspace}bpaceCommonWidgetRootId</tns:id>
<tns:type>{com.ibm.bspace}bpaceCommonWidgetRootId</tns:type>
<tns:version>1.0.0.0</tns:version>
<tns:url>/BusinessSpace/</tns:url>
<tns:description>Location of Business Space Common
Widgets</tns:description>
</tns:Endpoint>

<tns:Endpoint>
<tns:id>{com.ibm.bspace}bpaceWidgetRootId</tns:id>
<tns:type>{com.ibm.bspace}bpaceWidgetRootId</tns:type>
<tns:version>1.0.0.0</tns:version>
<tns:url>/BusinessSpace/</tns:url>
<tns:description>Location of Business Space Widgets</tns:description>
</tns:Endpoint>

<tns:Endpoint>
<tns:id>{com.ibm.bspace}bpaceServerRootId</tns:id>
<tns:type>{com.ibm.bspace}bpaceServerRootId</tns:type>
<tns:version>1.0.0.0</tns:version>
<tns:url>/BusinessSpace/</tns:url>
<tns:description>Location of Business Space Server</tns:description>
</tns:Endpoint>

<tns:Endpoint>
<tns:id>{com.ibm.bspace}bpaceWidgetHelpRootId</tns:id>
<tns:type>{com.ibm.bspace}bpaceWidgetHelpRootId</tns:type>
<tns:version>1.0.0.0</tns:version>
<tns:url>/BusinessSpaceHelp/</tns:url>
<tns:description>Location of Business Space Widgets
Help</tns:description>
</tns:Endpoint>
</tns:BusinessSpaceRegistry>
<!-- END NON-TRANSLATABLE -->

```

17. Press **Ctrl + S** to save the file. Then close the file.

Section B – Packaging the widget

We could use embedded scripts in IID to package up the artifacts. Instead we will export things to a folder structure as recommended by the BPM 7.5 product infocenter.

Step 7: Create the directory structure

The product infocenter suggests creating the following directory structure:

- `ear\widgets_name.ear` (one or more EAR files)
- `catalog\catalog_name.xml`
- `help\eclipse\plugins*`
- `endpoints\endpoints_name.xml`

18. Create a top-level directory named as the widget – HelloWorldWidget
 - a. Then create the 4 subdirectories as listed above. You may choose not to create the help sub-folder because in this example we haven't create any help files. But, it is also okay to have empty folders.

Step 8: Export the iWidget artifacts

Now we can export the required widget artifacts to the appropriate directories that were created.

19. In the **Enterprise Explorer**, highlight **HelloWorldWidgetEAR** project, right-click and select **Export->EAR file**.
 - a. Enter a destination (`.../HelloWorldWidget/ear/HelloWorldWidget.ear`)
 - b. Under Target runtime, choose to optimize for a specific server runtime, namely **IBM Process Server v7.5**
 - c. You may choose to export source files
 - d. Click **Finish**

Tip: For Business Space, the naming convention for the EAR is `widgetName.ear`. You might want to change the name of the exported EAR file from `HelloWorldWidgetEAR.ear` to `HelloWorldWidget.ear`.

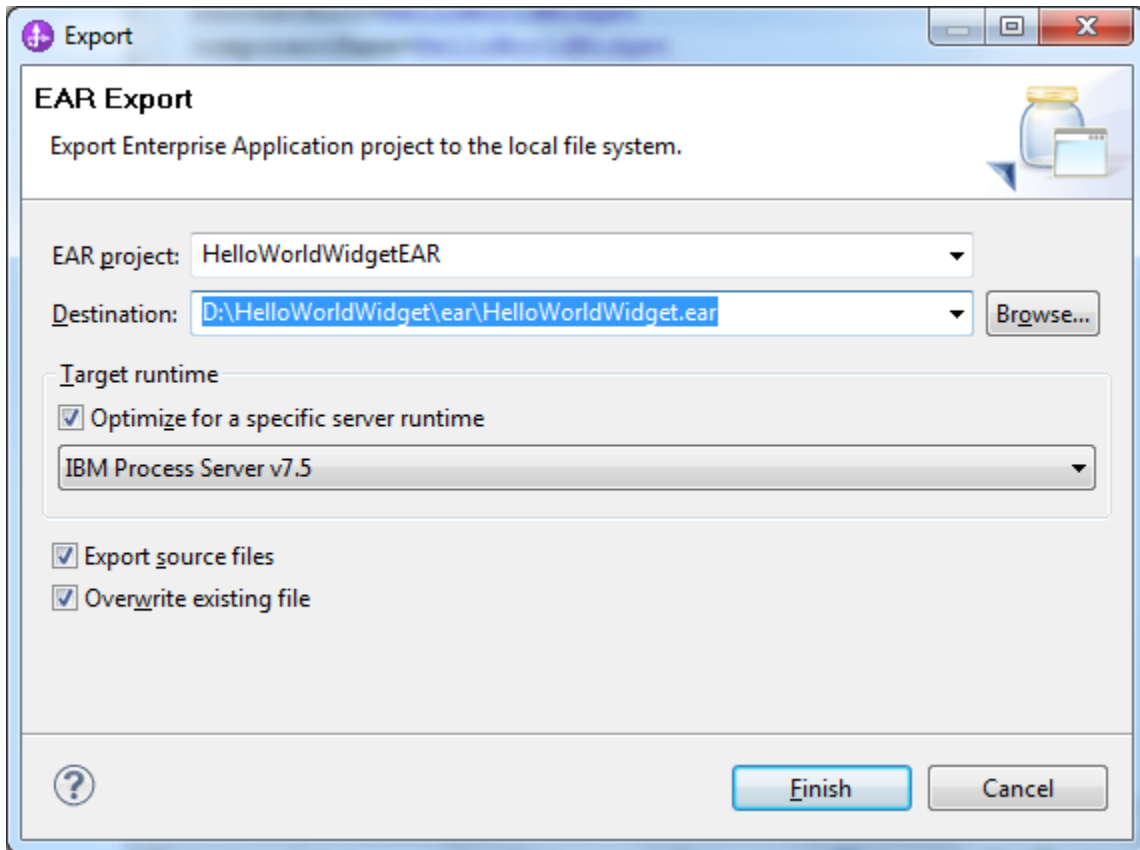


Figure B1

20. Back in the **Enterprise Explorer**, highlight **catalog_HelloWorld.xml** file, right-click and select **Export->File System**.
 - a. Click **Next**
 - b. Specify the target directory (.../HelloWorldWidget/catalog)
 - c. Click **Finish**

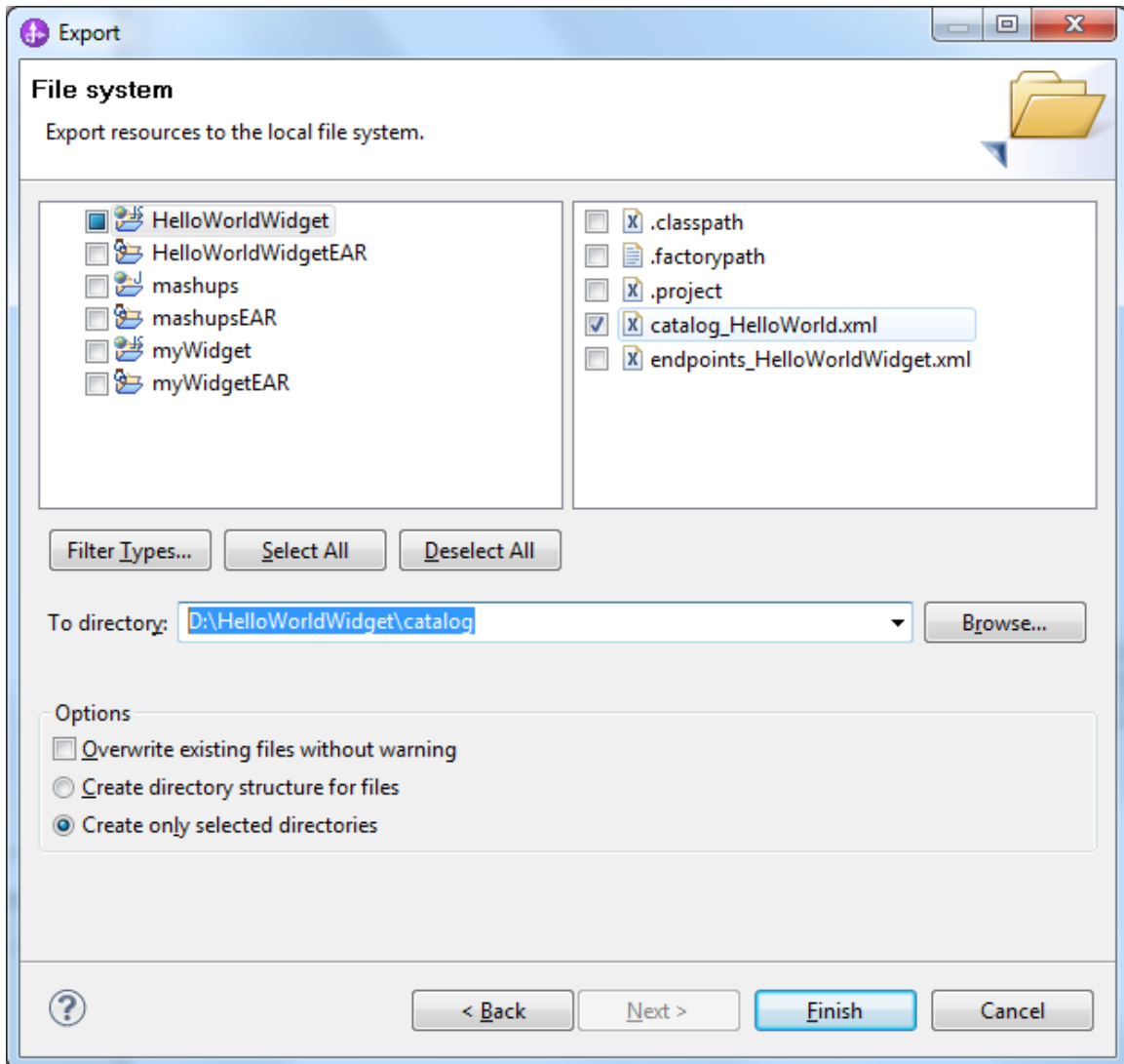


Figure B2

21. Similarly export **endpoints_HelloWorldWidget.xml** file to its target directory (.../HelloWorldWidget/endpoints)
 - a. Click **Finish**
22. The last step is to compress the ear, catalog, endpoints, and help directories into *WidgetName.zip*. The folder structure with the zip file is shown in Figure B3.

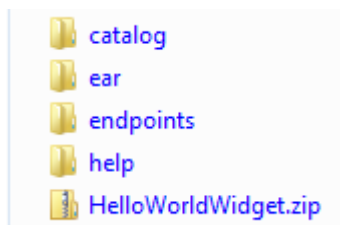


Figure B3

Section C – Deploying the widget

The IBM BPM Server need not be running for this operation because we will use the wsadmin script to install the widget in a non-connected mode.

Step 8: Use a script to install the iWidget

Use the wsadmin script to install/uninstall widgets.

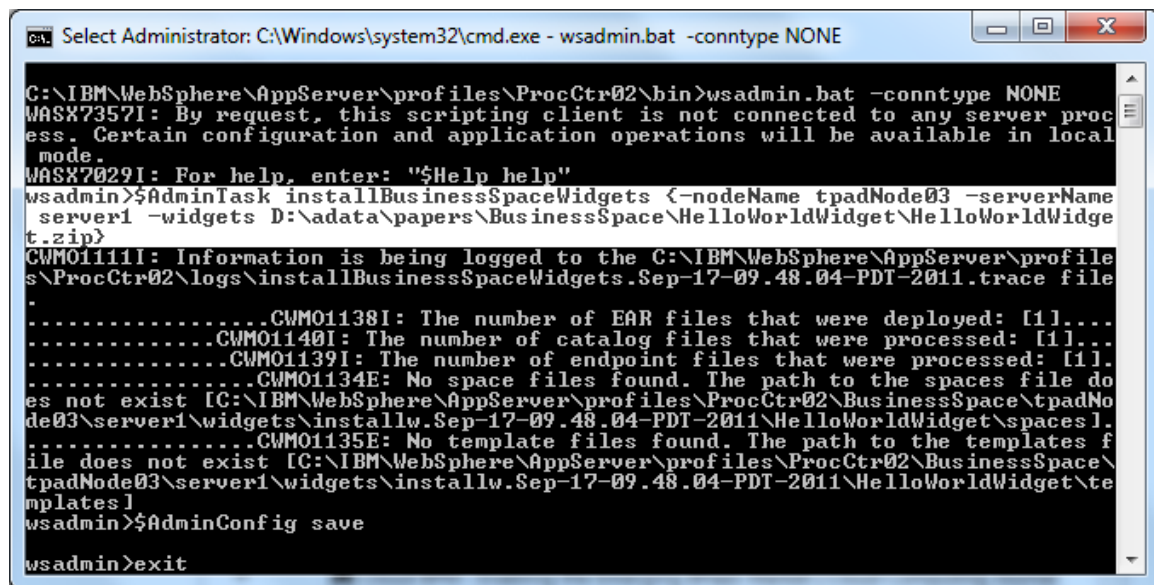
23. In a command window, change directories to <BPM_HOME>/<PROFILE_NAME>bin.
 - a. Enter `wsadmin.bat -conntype NONE` and then enter the appropriate command.
 - b. You may use Jython or Jacl command to install the widget (*installBusinessSpaceWidgets*). In this case, the Jacl command was used.

```
$AdminTask installBusinessSpaceWidgets {-nodeName node -serverName server -widgets fullpath}
```

- c. *Fullpath* is the name and location of the .zip file that was created.
- d. Figure C1 shows the command that was used.
- e. Remember to follow it with *AdminConfig save* command.

```
$AdminConfig save
```

- f. Finally *exit* the wsadmin command.



```
ca. Select Administrator: C:\Windows\system32\cmd.exe - wsadmin.bat -conntype NONE
C:\IBM\WebSphere\AppServer\profiles\ProcCtr02\bin>wsadmin.bat -conntype NONE
WASX7357I: By request, this scripting client is not connected to any server process. Certain configuration and application operations will be available in local mode.
WASX7029I: For help, enter: "$Help help"
wsadmin>$AdminTask installBusinessSpaceWidgets {-nodeName tpadNode03 -serverName server1 -widgets D:\adata\papers\BusinessSpace\HelloWorldWidget\HelloWorldWidget.zip}
CWM01111I: Information is being logged to the C:\IBM\WebSphere\AppServer\profiles\ProcCtr02\logs\installBusinessSpaceWidgets.Sep-17-09.48.04-PDT-2011.trace file
.....CWM01138I: The number of EAR files that were deployed: [1]...
.....CWM01140I: The number of catalog files that were processed: [1]...
.....CWM01139I: The number of endpoint files that were processed: [1]...
.....CWM01134E: No space files found. The path to the spaces file does not exist [C:\IBM\WebSphere\AppServer\profiles\ProcCtr02\BusinessSpace\tpadNode03\server1\widgets\installw.Sep-17-09.48.04-PDT-2011\HelloWorldWidget\spaces].
.....CWM01135E: No template files found. The path to the templates file does not exist [C:\IBM\WebSphere\AppServer\profiles\ProcCtr02\BusinessSpace\tpadNode03\server1\widgets\installw.Sep-17-09.48.04-PDT-2011\HelloWorldWidget\templates].
wsadmin>$AdminConfig save
wsadmin>exit
```

Figure C1

24. It is recommended that you restart the BPM Server.

Tip: To uninstall a widget the command is ***uninstallBusinessSpaceWidgets***. The syntax is the same as that for *installBusinessSpaceWidgets*.

Section D – Display the widget

The final step is to view the widget in Business Space.

Step 8: Use the widget in Business Space

Verify that the widget shows up in the widget palette in Business Space, then add it to a page.

25. After the BPM Server has restarted, launch **Business Space** (http://<HOST_NAME>:<PORT>/BusinessSpace).
 - a. Log in and go to a page in any Business Space and click **Edit Page**.
 - b. In the widget palette you should see the new widget named **Hello World**.

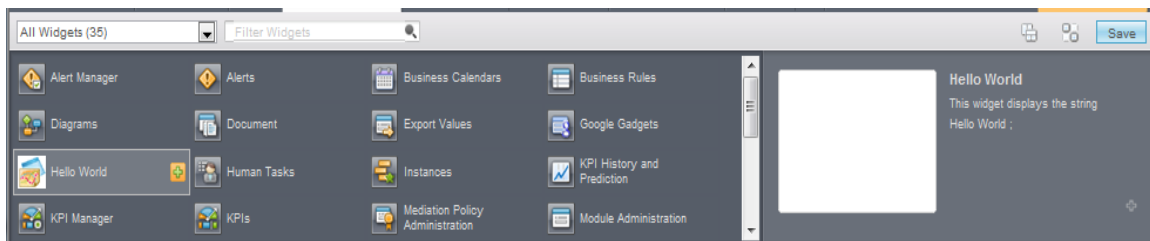


Figure C1

- c. Click the + sign next to it to add the widget to the page. The resulting widget should look like Figure C2.
 - d. Remember to click **Save** and then click **Finish Editing**.



Figure C2

Tip: If you encounter 404 catalog errors in Business Space when trying to add the widget to a page make sure the `iWidgetName.xml` definition URL in the catalog file is accessible. For example, take the definition from the catalog file in Section A and bring it up in a Web browser (http://<SERVER_NAME>:<PORT>/HelloWorldWidget/iWidget/widgets/HelloWorldWidget/HelloWorldWidget_iWidget.xml).

That should be it.

Conclusion

With the help of the procedures in the product infocenter, this document was created to provide a concise set of steps to create and deploy a simple widget in Business Space. IBM Integration Designer (IID) is one of many tools you can use to create an iWidget. And it is just as easy to create an event subscriber widget or an event publisher widget.

References

Business Space InfoCenter

http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5mx/index.jsp?topic=%2Fcom.ibm.wbpm.bspace.dev.doc%2Fdoc%2Fdeveloping_widgets%2Fcreatingwidgets.html

Business Space InfoCenter

http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5mx/index.jsp?topic=%2Fcom.ibm.wbpm.bspace.ic.main.doc%2Fbpace_welcome.html

WebSphere BPM InfoCenter

<http://publib.boulder.ibm.com/infocenter/dmndhelp/v7r5mx/index.jsp>

Note: The utility loader file can be found in
<BPM_HOME>/profiles/<PROFILE_NAME>/installedApps/<CELL_NAME>
/BspaceEAR_<NODE_NAME>_<SERVER_NAME>.ear/Bspace.war/com/ib
m/bpace/common/util/widget folder.
