



# Developing portlets using Eclipse and WebSphere Portlet Factory

Skill Level: Intermediate

Alex Barbosa Coqueiro (alexbc@br.ibm.com) IT Specialist, Lotus Technical Sales Team IBM

Helmar Martens (hmartens@us.ibm.com) IBM WebSphere Portlet Factory Worldwide Customer Support IBM

05 Jun 2006

This tutorial shows you how to develop a WebSphere portlet application for IBM WebSphere Portal V5.1 using Eclipse and WebSphere Portlet Factory. Developers at all skill levels (including those with novice Java, J2EE, and portlet development skills) can take advantage of the software automation technology offered by Portlet Factory to develop complex portlet applications. You see how to use WebSphere Portlet Factory to develop a portlet that retrieves and displays data from a database.

# Section 1. Before you start

Learn what to expect from this tutorial, and how to get the most out of it.

## About this tutorial

This tutorial shows you how to use WebSphere Portlet Factory's software automation technology to quickly develop complex portlet applications.

## Objectives

Specifically, you learn how to:

• Create and configure a Portlet Factory project.

- Create a new Portlet Factory model
- Add specialized database Builders to the model
- Use Builders to externalize the data
- Deploy the portlet to the portal

# Prerequisites

You should have at least introductory level skill in Java, J2EE, and portlet development.

## System requirements

This tutorial assumes that you have installed and configured WebSphere Portal Version 5.1x and one of the editions of WebSphere Portlet Factory 5.12.0 with Eclipse. If you want to use Rational Application Developer see the Portlet Factory product documentation. Portlet Factory is a plug-in which you install into Eclipse.

If you have the products, but have not already installed Portlet Factory, you can follow the instructions in Appendix A on page .

You also need a data source named jdbc/v5/sample which must be configured to connect to the Sample database that ships with DB2.

Before you start this tutorial:

- 1. Test your data source and make sure that you can connect to the Sample database.
- 2. Make sure that your WebSphere Portal server is up and running.

# Section 2. Develop a portlet application

### Introduction

WebSphere Portlet Factory provides a framework and development environment for developing portlets and Web applications. You can use it to rapidly develop portlets, even if you have a limited knowledge of the Java programming language.

WebSphere Portlet Factory supports a service-oriented architecture (SOA) by

implementing different application layers through the use of multiple models. Service Provider Models access back-end data and services. Service Consumer modes (presentation layer) provide a front end web user interface. Portlet Factory automatically generates both layers and creates all the necessary artifacts to enable loose coupling between layers.

Portlet Factory accomplishes software automation through its Builder technology. Builders are reusable software components which generate software assets, such as HTML, JSP tags, XML schemas, Java code, and other artifacts. For example, you can use a Builder to create a button on a JSP page, or to create a set of JSPs pages which display database content. Every Builder can be accessed and reused by other Builders to create new functionality.

Builder calls are managed and stored into models. In other words, a model is a container for Builder calls, and a model is an XML document which saves Builder calls and configurations.

You can deploy Portlet Factory projects to both WebSphere Application Server and WebSphere Portal Server. While developing Portlet Factory projects as portlet applications, you have to take into consideration a few aspects that are specific to portlet application development.

## Create a new portlet application

You could create a portlet application using Rational Application Developer. The process described in this article is very similar to work with Application Developer. See the installation instructions for more information.

The scenario for this tutorial is a common development scenario in which WebSphere Portal and Eclipse are installed on the same machine. The steps for creating a project to be deployed to a remote server are slightly different from the ones describe here. You could also choose to create a project which uses TomCat as the development server; however, these other scenarios are not covered.

You begin by opening Eclipse.

- In the Windows Start menu, select Start => All programs => IBM WebSphere => Portlet Factory => Designer.
- 2. Select your workspace, and click **OK**. Now, you create a Portlet Factory project.
- 3. From the menu select **File => New => Factory WebApp Project**.

Figure 1. Create a Factory WebApp Project

X

Factory Models - Eclipse Platform									
File	Edit	Source	Refactor	Navigate	Search	Project	Run	Window	Help
New Alt+Shift+N		Project							
(	ilose Thee A		Ctrl+f	F4 Shift+E4	👸 Fact	tory Web/	App Pro	oject	

- 4. You see the first page of the Project creation wizard.
- 5. For Project name enter DemoPortlet.
- 6. Leave Use default checked, and click **Next**.
- 7. On the next screen, the Add Features screen, do not select any feature; instead just click **Next**. The Project Server Target screen displays.

🚝 Create WebApp Project **Project Server Target** Specify application server target for project. Server Type WebSphere5 Factory Development WAR Location Installed Applications Directory C:\WebSphere\AppServer\installedApps\pvcvm1 Choose Application Name DemoPortlet Automatic Deployment Automatically Deploy Project to WAS Server WAS Server To Deploy To WebSphere\_Portal -Server Host localhost Server Port 9081 C J2EE 1.4 < Back Next > Cancel

Figure 2. Project Server Target

- 8. In the Project Server Target window, enter the following information:
  - Server Type: WebSphere5
  - Installed Application Directory: >AppServerRoot<\installedApps\>node< (this is the location of the node folder in the installedApps directory of your local WebSphere Portal Server installation AppServer folder).
  - Application Name: DemoPortlet

- Automatic Deployment:
  - 1. Check the Automatically Deploy Project to WAS Server box
  - 2. WAS Server to Deploy To: WebSphere\_Portal
- Make sure that the Server Host and Server Port inputs reflect your local WebSphere Portal configuration. Because you are deploying the WAR file to the development portal, use the default port which is 9081.
- 9. Click **Next** to go to the Java Settings page.
- 10. Accept all the default values, and click **Next**. You see the WebSphere Portlet WAR Settings page, as shown in Figure 3.

Figure 3. WebSphere Portlet WAR settings

hSnhere Portlet	WAR Settings	-	
ecify name and location	on for Portlet WAR file.	505	
VebSphere Portlet WA	R		
Create WebSphere	Portlet WAR		
Server is Available			
WP Root	C:\WebSphere\PortalServer	Choose	
PB Portlet Jar Location	pps\Abstract B_r Portlets_PA_1_0_46.ear\BOBuilderPortlet.war\WEB-INF\lib	Choose	
Portlet WAR Name	PF4WS-DemoPortlet.war		
Portlet WAR Location	C:\WebSphere\PortalServer\installableApps	Choose	
Admin URL	http://localhost:9081/wps/config		
Admin User   wpsadmin			
Admin Password			
Admin Password Test Admin Creden	tials		
Admin Password Test Admin Creden	tials		

- 11. On the WebSphere Portlet WAR Settings screen, provide the following inputs.
  - Check the box for Create WebSphere Portlet WAR.
  - Check the box for Server is Available.

- WP Root: Enter C:\WebSphere\
- PB Portlet Jar Location: Navigate to the installedApps directory, and choose an application that contains a pbportlet.jar file <Portal</li>
   Server root>\installedApps\Abstract
   B\_rPortlets\_PA\_1\_0\_46.ear\BOBuilderPortlet.war\WEB-INF\lib
- Portlet WAR Name: Accept the default value, PF4WS-DemoPortlet.war
- Portlet WAR Location: Specify the path to the installableApps folder in the local WebSphere Portal installation. For example: <Portal Server root>\installableApps\
- Check the box for Automatically Deploy Portlet WAR to WebSphere Portal (Version 5.x Only).
- Admin URL: The default for local portal is http://localhost:9081/wps/config
- Provide the proper WebSphere Portal admin credentials, and test the credentials to make sure that you can connect to the portal from your project.
- 12. Click Next to go to the JRS 168 Portlet WAR Settings.
- 13. In the JSR 168 Portlet WAR Settings do not check **Create JSR 168 Portlet WAR** box.
- 14. If you are presented with the Portal Admin Information Page, make sure that the default values reflect your environment, and then click **Next**.
- 15. In the Summary and Important Details screen, confirm that the information presented is correct, and click **Finish**.
- 16. If you are prompted to add jars from the selected Feature, set to the project Java build path, and then click **Yes**.

When this task completes, both the Dev and Portlet WAR files (in this tutorial, PF4WS-DemoPortlet.war) are created and, in this example, are deployed to the server. The Dev WAR is the one that is invoked when you run the application from the Designer.

After Eclipse has finished creating your project, you should have no error messages in your workspace. If you see error messages, check the paths, portal settings, and credentials to install the portlet, and repeat the deploy process.

### Create the portlet to access the database

Now that you have the project, you create the portlet to access the database.

- 1. From your workspace, right-click on the root directory of your project, and select **New => Factory Model**.
- 2. On the Choose Project screen, select **DemoPortlet**, and then click **Next**.
- 3. Leave Select Model empty (blank), and click Next.
- 4. Set the Page Type to Imported Page, and click Next.
- 5. For the Model Name, specify DemoPortletTest.
- 6. Click **Finish**.

Now the DemoPortletTest model is active in your workspace.

## Connecting to a database

The Portlet Factory perspective is divided into four panels, including:

- 1. The top left panel, which presents the Navigator and contains your project and all the artifacts included in your project.
- 2. The bottom left panel, titled Outline, which contains the Builder call list.
- 3. The top right panel, which contains the space that hosts the WebApp Tree the Model XML, the Builder Call Editor, and the WebApp Diagram.

To connect to a database, you add a Builder to the model.

1. From the Outline, open the Builder palette (which contains the list of all Builders calls) by clicking on the icon in the middle left portion of the Outline screen that looks like a gear with a cross in front of it (as shown below). When you hover the mouse over this icon the context message reads Add a Builder Call to the Current Model.

E Outline 🛛		(€) ▽ □ □
#	Name	Туре

- 2. From the Builder Palette, select **Data integration** for the category, and then **SQL Call** for the Builder type.
- 3. Click **OK**. The Builder Call editor opens to display the SQL Call Builder, as shown in Figure 4.

Figure 4. SQL Call Builder call

Concentral 12					
cho name> 25	1				
Fetch DataSource Name	s				
SQL DataSource:	ð	)dbc/v5/Sample			
Database Explore	r				
SQL Statement:	ð	SELECT EMPNO, F SALARY AS "Se	"IRSTNME, LASTNAME alary" from D82ADMIN	WORKDEPT, PHONENO, BONJ EMPLOYEE	15, COMM,
Parameter Binding *:	0	Manual ⊂ Aut     Aut	omatic (Create Distinc	Variables) 🦳 Automatic (Cre	ate IML Variable)
Parameters:	ð	P Position	Direction	308C Type Cast	Value or Set/Cast Method
<ul> <li>ResultSet Handlin</li> </ul>	g				
These inputs govern both statement or an STP invoc	the char- ation the	acteristics of a Res it returns a ResultS	ultSet returned by the Set, then make sure th	SQL call and the translation re e builder is configured to trans	equirements for ResultSet data to XML or schema t form the ResultSet into XML.
Concurrency:	31	Read Only		-	
Scroll Type:	3	Forward		-	
Transform Result:	C None Complete XML Document C Paged XML DataRetriever				
Top Element Name:	ð	Employees			
Row Element Name:	ð	Employee			
		e	Sen @ From SCL S	Latement C From Alternate S	OL Statement C Use Existing URL/File/Variable
Schema Generation:	ð	Comit Generation	in step in them side s		

4. Provide the following inputs to this Builder call:

Field	Setting
Name	GetSummaryData
Fetch Data Sources	jdbc/v5/sample
SQL statement	SELECT EMPNO, FIRSTNME, LASTNAME, WORKDEPT, PHONENO, BONUS, COMM, SALARY AS "Salary" from DB2ADMIN.EMPLOYEE
Concurrency	Read Only

Scroll Type	Forward
Transform result	Complete XML Document
Top Element	Employees
Row Element	Employee
Schema generation	From SQL Statement
Schema Regen Time	Only When SQL Builder Changes Are Detected

5. Click OK.

**Troubleshooting**: If the Problems tab displays either one or both of the following error massages, then either you do not have a valid connction to the database from your model or this user does not have the permissions to access this table. By default, this table belongs to the ADMINISTRATOR schema.

Database metadata generation on the server failed Did not receive and XML Schema from the server...

You might need to change the schema from DB2ADMIN.EMPLOYEE to ADMINISTRATOR.EMPLOYEE.

6. WebSphere Portlet Factory provides a JSP page to test your data source connection. It is located in the factory\util directory of the servable content of your project. You can invoke this page by entering the following URL into your browser:

http://localhost:9081/<your\_project\_Name>;/factory/util/datasourcetest.jsp

7. Save your model.

### Add a View & Form Builder to expose the data

The View & Form Builder adds functionality for displaying data and for adding, updating, and deleting records from a back-end system. This Builder can either create all the JSP pages and map the values from the source to the page, or it can map the values to existing HTML pages.

To add the View & Form Builder:

- 1. Open the Builder palette.
- 2. For for the category, select Data Integration.
- 3. From the Builder type panel, select View & Form.

Figure 5. View page options section of the View & Form Builder call

Properties			•
Name *:	j Summary		1
- View page options	s		
view Method *:	j GetSummaryDataInvoke		
View Variable *:	J Variables/GetSummaryDataTransformVini		
View Page HTML *:	j //actory/pages/view_and_form_view.html		
HTML Template File:	j /factory/html_templates/default.html		
Back Button Text:	) Back		
View Page Type:	Create fields from view variable schema C Use fields from imported HTML	_	
Paged Data Display:	V 6		ļ
Rows Per Page:	ğ 10		
- Input page option	15		
Create Input Page:	<u>گ</u> [		'
Enable Input Validation:	<u>ð</u>		
- Row Details Supp	ort		
Create Link To Details:	ALC: ALC: ALC: ALC: ALC: ALC: ALC: ALC:		1

4. Provide the following inputs for this Builder call:

Field	Setting
Name	Summary
ViewMethod	GetSummaryDataInvoke Click on

	the chooser button to open a list of available methods (that is, the button with ellipses to the right of the input field). The GetSumr method was created by the SQL Call Builder. This method gets invoked when you want to execute the SQL statemen you provided in the SQL Call Builder call.	naryDataInvoke
View variable	Use the chooser to select Variables	s/GetSummaryDataTransformXML
View Page HTML	/factory/p	ages/view_and_form_view.html
HTML Template File	/factory/h	tml_templates/default
Back Button Text	Back	

View Page Type	Create Fields from view variables schema
Paged Data Display	Checked
Rows per Page	10

- 5. Leave both inputs located in the Input page options section blank.
- 6. Expand the Row Details Support section of the View & Form Builder call (shown in Figure 6).

Figure 6. Row Details Support section of the View & Form Builder call

Johnnory (2)		1
Rows Per Page:	j 10	
<ul> <li>Input page option</li> </ul>	s	
Create Input Page:	<u>ð</u>	
Enable Input Validation:	ů C	
- Row Details Suppo	ort	
Create Link To Details:	N C	
Details Link Column:	Ú EMPNO V	
Details Link Text:	(Variables/EmployeeLoopVar/Employee/EMPNO)	
Details Action Type:	Get details data directly from selected row	
Details Page HTML:	jfactory/pages/view_and_form_view.html	
HTML Template File:	jfactory/html_templates/default/html	
Back Button Text:	j Back.	
Details Page Type:	Create fields from view variable schema C Use fields from imported HTML	
Label Translation	and Sample HTML	
- Advanced		
Generate Main:	V 6	
<		2
	OK Cancel Apply	Help

7. Enter these settings:

Field	Setting	
Create Link to Details	Checked	
Details Link Column:	EMPNO	
Details Link Text	Use the chooser to select \${Variabl	es/EmployeeLoopVar/Employee/EMPNO}
Details Action Type	Get details data directly from selected row	
Details Page HTML	/factory/p	ages/view_and_form_view.html
HTML Template file	/factory/h	tml_templates/default.html
Back button text	Back	
Details Page Type	Create fields from view variable schema	

- 8. Leave the other option as their default values.
- 9. Scroll down to the **Advanced** section of the Builder located at the bottom of the Builder call. Expand this section and check **Generate main**.
- 10. Click **OK** and save your model.

### Run your model

After you finish the development process, the next step is to test your components. The WebSphere Portlet Factory provides an option to run your model to test the components that you have created.

1. Create a configuration to run your model.

## Figure 7. Running configuration menu option



2. Click on the down arrow in the tool bar, and select the **Run** option.

. Create, manage, and run configurations Configurations: Name: Training Eclipse Application St Factory Model Main Browser Tracing Common Training 3 Java Applet 🗾 Java Application Model to run Ju JUnit run model from current active model editor (with applied profiles) Junit Plug-in Test SWT Application run named model G C run with Default profile(s) alway Run URL http://localhost:9081/HM\_Training\_DF/webengine/DemoPortletTest Delete Run Close

Figure 8. Run configuration screen

 In the Run configuration screen, select Factory Model, click New, provide a name, and then click Run. The Employee's View displays, as shown in figure 9.

Figure 9. Employees View

http:/	/localhost	:9081/HM_	Training	DF/weber	igine/De	moPort	letTest -	Micro	soft Int	ernet Explore
le Edi	t View F	avorites To	ools Help							
Back	• 🕤	- 🖹 💈	1 🏠 🔎	Search	📌 Fav	orites	🔊 Media	Ø	8.	è 🖻
ddress	http://loc	alhost:9081/H	M_Training_D	F/webengin	e/DemoPo	ortletTest				
EMPIIO	FIRSTHME	LASTNAME	WORKDEPT	PHONENO	BONUS	COMM	Salary			
000010	CHRISTINE	HAAS	A00	3978	1000.00	4220.00	52750.00			
000020	MICHAEL	THOMPSON	B01	3476	800.00	3300.00	41250.00			
000030	SALLY	KWAN	C01	4738	800.00	3060.00	38250.00			
000050	JOHN	GEYER	E01	6789	800.00	3214.00	40175.00			
000060	IRVING	STERN	D11	6423	500.00	2580.00	32250.00			
000070	EVA	PULASKI	D21	7831	700.00	2893.00	36170.00			
000090	EILEEN	HENDERSON	E11	5498	600.00	2380.00	29750.00			
000100	THEODORE	SPENSER	E21	0972	500.00	2092.00	26150.00			
000110	VINCENZO	LUCCHESSI	A00	3490	900.00	3720.00	46500.00			
Automatical Strength				0407	600.00	2340.00	29250.00			

4. Click on the **EMPNO** link for the first record to see the details for that record.

Figure 10. Employee Detail

🔄 http://loca	lhost:9081/HM_Training_DF/webengine/DemoPortletTest/Action!_gen_call_Summary_SelectR = A
File Edit Vie	w Favorites Tools Help
G Back •	🕑 - 🖹 🗟 🏠 🔎 Search 🤺 Favorites 🔇 Media 🤣 🍰 🛁
Address 🕘 http	o://localhost:9081/HM_Training_DF/webengine/DemoPortletTest/Action1_gen_call_Summary_SelectRow?objTypes_SelectRo
EMPNO	000010
FIRSTNME	CHRISTINE
LASTNAME	HAAS
WORKDEPT	A00
PHONENO	3978
BONUS	1000.00
COMM	4220.00
Salary	52750.00
Back	

# Deploying the portlet

To make this model a portlet, you add a Portlet Adapter Builder.

From the Builder Palette:

1. For the category, select Portlet Integration; for the Builder type, select

Portlet Adapter.

- 2. Click **OK**. The Builder Call Editor opens to display the Portlet Adapter Builder call.
- 3. Specify these settings:
  - Name: DemoPortletTest
  - Portlet Title: WebSphere Portlet Factory Demo
- 4. Click **OK**.
- 5. Save the model by clicking on the Save icon.

Now, you need to rebuild the portlet WAR file. Right-click on the project name, and then choose **Rebuild WAR => Rebuild Portlet WARs**.

Because you enabled auto deploy and auto refresh when you created the project, Portlet Factory deploys and updates the WAR file for you. This is just one more way that Portlet Factory helps you reduce the time it takes to deploy a portlet application.

To confirm that your portlet has been deployed to your local portal:

- 1. Login to WebSphere Portal as the portal administrator.
- 2. Click Administration => Portlet Management => Applications.
- 3. Search by selecting **Title starts with**; search for **DemoPortletTest**. If English is not your default locale language, you might need to search by **Last Modified** instead of **Title starts with**.
- Click on this application, and confirm that your WebSphere Portlet Factory Demo portlet is available.
   Troubleshooting: If you do not see the DemoPortletTest in the list of portlets, you might not have saved the DemoPortletTest Portlet Factory Model before rebuilding the Portlet WAR. Go back to Eclipse, save the model, and then rebuild the Portlet WAR.
- 5. Place the Portlet on a test page using the WebSphere Portal Server Edit Layout Process.

View your portal which should look similar to Figure 11.

Figure 11. The WebSphere Portlet Factory Demo accessed from the portal.

le Eak	View Fav	orites Tools	Help					
子 Back	• 🕤 •	🖹 💈 🦿	Sear	ch 🤺 Favo	orites 🍳	Media	<del>@</del> @-	ا 🍣
/ddress 🕻	http://localho	ost:9081/wps/my	portal/!ut/p/_s	.7_0_A/7_0_V	9/.cmd/ad/	.ar/206242	8155/.c/6_0_	CH/.ce/7
WebSphere	Portal							1
had a set of second								
< ►	Content Pu	blishing Doc	uments We	lcome My	Work 1	rourCo Fi	nancial My	Finance
Training	g My Page	Demo						
WebSphe	ere Portlet Fa	ctory Demo						
EMPNO	FIRSTNME	LASTNAME	WORKDEPT	PHONENO	BONUS	COMM	Salary	
000010	CHRISTINE	HAAS	A00	3978	1000.00	4220.00	52750.00	
000020	MICHAEL	THOMPSON	B01	3476	800.00	3300.00	41250.00	
000030	SALLY	KWAN	C01	4738	800.00	3060.00	38250.00	
000050	JOHN	GEYER	E01	6789	800.00	3214.00	40175.00	
000060	IRVING	STERN	D11	6423	500.00	2580.00	32250.00	
000070	EVA	PULASKI	D21	7831	700.00	2893.00	36170.00	
000090	EILEEN	HENDERSON	E11	5498	600.00	2380.00	29750.00	
-	THEODORE	SPENSER	E21	0972	500.00	2092.00	26150.00	
000100		LUCCHESSI	A00	3490	900.00	3720.00	46500.00	
000100 000110	VINCENZO							

# Conclusion

In this tutorial, you learned how to create a portlet, using WebSphere Portlet Factory, that runs on WebSphere Portal and accesses a database. You saw how Portlet Factory automates the creation of portlet artifacts such as JSP pages, XML schema, Java code, and so on. You also saw how Portlet Factory takes care of the deployment and updates of the WAR file.

# Appendix A: Installing WebSphere Portlet Factory

To install the WebSphere Portlet Factory plug-in, contact your local IBM sales office to get the installation files. For more information, see How to buy WebSphere Portlet Factory.

- 1. Start the WebSphere Portal server.
- 2. Extract the zip installation files.
- 3. Run Factory.exe.

- 4. On the Introduction page, click **Next**.
- 5. On the License Agreement page, select a License language, and click **Next**.
- 6. After you read and accept the terms of the License Agreement, click **Next**.
- 7. Choose an install location. For example, accept the default location (C:\Program Files\IBM\WebSphere Portlet Factory), and then click **Next**.
- 8. Specify install items. For example, to follow along in this article, choose the third option, **And Eclipse**. If you want to install Portlet Factory with the IBM Rational Application Developer environment, read the install.htm in the zip installation files.
- 9. Click **Next**.
- 10. To specify the IDE, accept the default location (C:\Program Files\eclipse3.1.1).
- 11. Click Next.
- 12. Click **Install**. Although the installation graphic indicates it is installing IBM Workplace Dashboard Framework, the WebSphere Portlet Factory is being installed as well. When the installation is complete, click **Done**.
- Start Eclipse from either the Windows Start menu at IBM WebSphere => Portlet Factory => Designer, or by using Windows Explorer to navigate to C:\Program Files\eclipse3.1.1\eclipse.exe.
- 14. When prompted, select the default workspace by clicking **OK**.

# Resources

### Learn

- WebSphere Portlet Factory product documentation. Provides access to all current product documentation including InfoCenters, release notes, and readmes for all releases of WebSphere Portlet Factory.
- WebSphere Portal product documentation. Provides access to all current product documentation including InfoCenters, release notes, and readmes for all releases of WebSphere Portal.
- Developing portlets which access SAP R/3, using Bowstreet Portlet Factory and WebSphere Studio: This two-part tutorial steps through developing a WebSphere portlet application which uses data from a SAP R/3 system. You use the Bowstreet Portlet Factory and IBM WebSphere Studio Application Developer (or WebSphere Studio Integration Edition) to create two portlets.
- Developing an On Demand Workplace, Part 12: Enable and customize employee self-service with Bowstreet Portlet Factory and PeopleSoft: Explore the Bowstreet Portlet Factory for WebSphere, a rapid application development platform for WebSphere Portal, and learn application integration with PeopleSoft HR portlet.
- developerWorks WebSphere Portal zone: Provides a wide variety of technical resources to help you develop portals and portlets.
- Stay current with developerWorks technical events and Webcasts.

### Get products and technologies

- Rational Application Developer V6: Download trial software from developerWorks. Includes the portal tools and a test runtime copy of portal that you can use to develop a prototype.
- Build your next development project with IBM trial software, available for download directly from developerWorks.
- How to buy WebSphere Portlet Factory

### Discuss

• Participate in developerWorks blogs and get involved in the developerWorks community.

# About the authors

#### Alex Barbosa Coqueiro

**Alex Barbosa Coqueiro** is an IT Specialist in IBM Lotus Software Group in Brazil. He has over ten years of experience in object-oriented development and is certified by Sun as an Architect (SCEA), Web Developer (SCWD) and Programmer (SCJP) for Java. His areas of expertise include WebSphere Application Server, WebSphere Portal Server.

#### Helmar Martens

**Helmar Martens** works as a Level 2 Support Engineer in Raleigh, North Carolina. Helmar has worked with WebSphere Portlet Factory since 2000. He has worked with Portlet Factory in the capacity of Quality Assurance Engineer, Developer, and currently supports this product.