Chapter XIII

Writing Portlets: A Quick Lesson for IT Managers

This section is intended for IT managers who need to understand the basics of developing portlets. We will take you briefly through the development cycle of a portlet: from writing the code, to compiling, up to deployment. You must start with environment set up and continue with writing, compiling, and testing the portlets.

The code example in Figure 110 shows a simple portlet called HelloMonashWorld class. This class inherits from the class PortletAdapter and processes the following methods:

- `init(portletConfig)` initializes the portlet and obtains its configuration, which is passed to the method in portletConfig object.
- `doView(PortletRequest portletRequest, PortletResponse portletResponse)` works in a similar fashion as the servlet’s `doGet()` method; it receives a user request and creates a new object in response.
The implementation of the initialization phase uses a portlet configuration object to load the initial configuration parameters (PortletConfig portletConfig).

You can see from the code that the portlet is using similar mechanics for writing output as those found in a servlet; in the doView() helper method, the portlet obtains the PrintWriter object and attaches this object to the response object:

```
   PrintWriter writer = portletResponse.getWriter();
```

Since we have used the PrintWriter object, we also have thrown the appropriate exception: IOException. When the rendering phase starts (page aggregation), the output stream created earlier is written to the servlet’s output.

Figure 111 shows the structure of portlet configuration APIs. The portal calls the service() method when the portlet is required to render its content. To enable portlets to use pluggable services via dynamic discovery, the Portlet API provides the PortletService interface. A PortletService is accessed from the PortletContext.getService() method, which creates the service and returns it to the portlet. A portlet service can be invoked only from within a portlet.

To compile portlet Java source code, the environment variables must be set up (namely CLASSPATH) so the java compiler can find all the *.jar files that are