Chapter VI

Java Server Pages (JSP)

Good Web application design always tries to separate business objects from controls and the presentation. The benefit of using Java Server Pages (JSP) is the ability to separate Web page design from functionality and content dynamics (coding). The main features of the JSP technology are:

1. Text-based documents written in a language that describes request processing and response construction;
2. Extensibility to define the tags and associate them with functionality; and
3. Ability to access objects on the server.

A JSP page is a text-based document that contains both static data as HTML, WML, SVG or XML, and JSP elements that provide dynamic content. The servlets include Java classes that generate dynamic Web HTML content using print() statements. The JSP technology redefined servlets, and it is now a key component of the J2EE specification. Many commercially available application servers support JSP (examples include BEA WebLogic, IBM WebSphere, Sun Microsystems Application Server 8). A JSP page is a traditional Web page with the HTML mark up and some “invisible” code hidden in the tag library. The prescribed file extension is .jsp, and the server provides a special processing mechanism for this.
When a JSP page is developed and deployed (Figure 61), it is translated into a Java Servlet Implementation (servlet java code), which is then compiled into the Java Servlet class. These two steps are typically completed offline during the deployment stage. However, online translation and compilation are also available. After deployment, the JSP page is called and passed directly to the processing stage where an instance of the servlet class (Servlet Object) is created. The Servlet Object produces the response, which is then returned to the client’s browser via the HTTP connection.

The basic components of the JSP system are:

- **Template Data**: Static elements written in HTML or XML.
- **Directives**: JSP elements that are interpreted at the translation time.
- **Scripting Elements**: JSP elements used to define a variable and/or method declaration, expressions, and blocks of commands (called scriptlets).
- **Action Elements**: Commands that look like tags but are implemented as Java code. Standard actions contain commands like forwarding the request to another resource, creating JavaBean objects, and so forth. In addition, custom actions can be designed via JSP tag extension support.