

# Oracle JDeveloper Release 9i Sneak Preview

*An Oracle White Paper  
May 2001*

# Oracle JDeveloper Release 9i Sneak Preview

## INTRODUCTION TO JDEVELOPER RELEASE 9i

Oracle JDeveloper is a J2EE(tm) development environment with end-to-end support for developing, debugging, and deploying e-business applications. JDeveloper empowers users with highly productive tools, such as the industry's fastest Java debugger, a new profiler, and the innovative CodeCoach tool for code performance analysis and improvement.

To take J2EE application development to a higher level of productivity, JDeveloper offers Business Components for Java (BC4J), a standards-based, server-side framework for creating scalable, high-performance Internet applications. The framework provides design-time tools and runtime services to drastically simplify the task of building and reusing business logic.

In this white paper, you will get a first glance at the upcoming 9i release and its most interesting new features. A beta release will be available this summer, with the production release expected later this calendar year.

## JDeveloper Covers the Complete Development Life Cycle

Java is a relatively new language and the development environments are catching up with traditional client/server tools in terms of completeness and richness. Initially a Java IDE had a visual editor, a code editor, a compiler, a debugger, and some wizards.

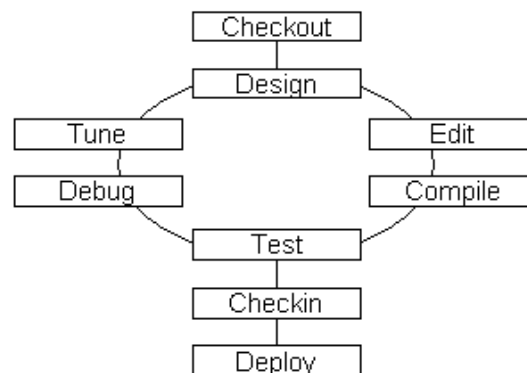


Figure 1. The complete life cycle for application development in JDeveloper

Today, the 9i release of JDeveloper provides complete support for the application development life cycle as depicted in Figure 1.

In a typical scenario, a developer launches JDeveloper, checks out an application from the source control system, and starts the development cycle. UML modelers help the developer with the design of the application and possibly with the generation of source code. JDeveloper provides wizards and editors – both visual and code centric – to add functionality and various tools to compile, test, debug, and tune the application. When satisfied, the developer can check the application back into the source control system and, after sufficient testing, deploy it to its final destination.

### 100% Java Version of JDeveloper Runs on Windows, Linux and Solaris

The 9i release of JDeveloper has been completely rewritten in Java. JDeveloper 3.x and earlier releases were a hybrid of Java and other programming languages, and run on Windows only.

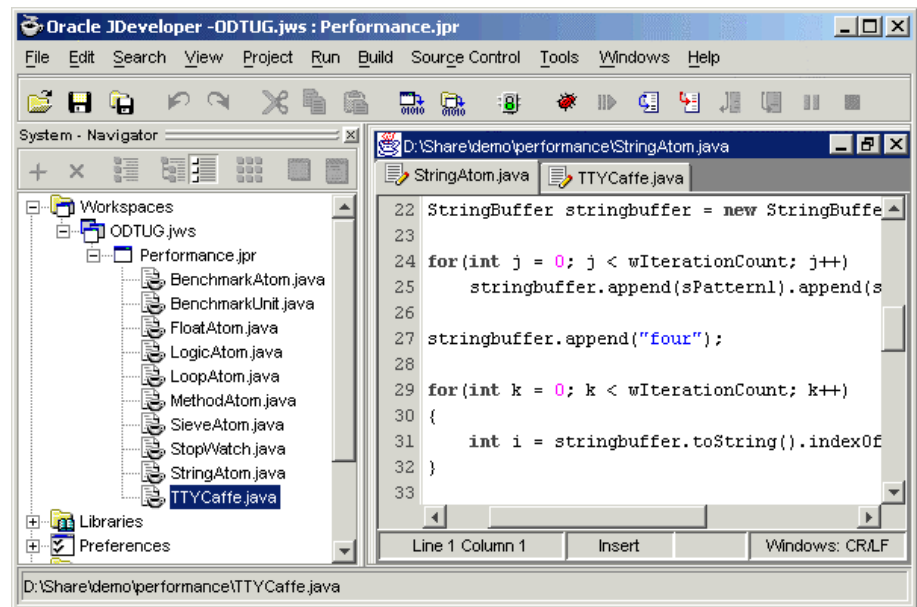


Figure 2. 100% Java version of Oracle JDeveloper

This release of JDeveloper runs on any platform that has a Java Virtual Machine (SDK 1.3 and later) and will be supported on Windows NT, 2000 and XP, Linux, and Solaris.

Another advantage is that the development environment is now fully extensible through the Addin API. JDeveloper is becoming the common development environment at Oracle, allowing internal development teams, 3<sup>rd</sup> party vendors, and customers to add functionality to the product.

JDeveloper is fully extensible through its Addin API, allowing anyone to add functionality to the product.

## DEVELOPMENT AND DEPLOYMENT OF J2EE APPLICATIONS

JDeveloper Release 9i fully leverages the J2EE platform and supports the latest standards, including EJB 1.1, JSP 1.1 and Servlets 2.2. Developers can choose from wizards, visual editors and code editors to build J2EE applications and to package them as standard archives that can be deployed on any J2EE server.

The JDeveloper debugger uses the standard JPDA protocol to enable remote debugging of Java code running on a J2EE server. See the debugger section below for operating system restrictions of the debugger.

### Oracle Business Components for Java

Sun Microsystems has created the J2EE Blueprints to help developers build robust J2EE applications. It provides a set of design patterns and samples. With these blueprints, developers typically write a lot of code to build a complete e-business application.

Oracle JDeveloper provides Business Components for Java, a J2EE framework that implements these design patterns. It offers tremendous productivity gains for developers by handling the Object/Relational mapping and the binding between the different tiers in the J2EE architecture. Business Components for Java provides both declarative and code-centric tools to define business logic, while it takes care of the underlying infrastructure.

Oracle Business Components for Java implements J2EE best practices and enables developers to productively deliver scalable e-business applications.

### Integration with Oracle9i Application Server

While JDeveloper supports the development, packaging and debugging of applications on any J2EE server, it makes deployment to Oracle9i Application Server extra productive by providing one-click automatic deployment from within the tool.

## JAVA ALONE IS NOT ENOUGH

Over the years, Java has become *the* programming language for the Internet, with tremendous mind share and millions of developers worldwide. Some of the reasons for this popularity are its platform independence, its simplicity, and its powerful component model.

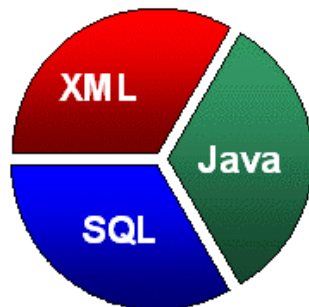


Figure 3. Java, XML and SQL – a killer combination

To build complete e-business applications, developers will need more than Java alone. Oracle believes strongly, and has invested heavily, in the combination of Java, SQL, and XML. Java is used for programming the business and presentation logic, SQL for interacting with the database, and XML for passing information between loosely coupled applications.

JDeveloper helps developers build e-business applications using Java, XML, HTML, SQL, and PL/SQL and provides various code editors and visual tools for each of these languages.

### XML tools

JDeveloper offers utilities to create, parse, generate and transform XML documents.

The Oracle XML Developers Kit (XDK) is integrated into JDeveloper, offering many utilities for Java developers to handle, create, and transform XML. For example, with the XSQL Servlet, developers can query and manipulate database information, generate XML documents, transform them using XSLT stylesheets, and make them available on the web.

JDeveloper Release 9i has a new schema-driven XML editor. An XML schema defines the structure of an XML document and is used in the editor to validate the XML and to help the developer when typing. This feature is called Code Insight and provides a list of valid alternatives for XML elements or attributes in the document.

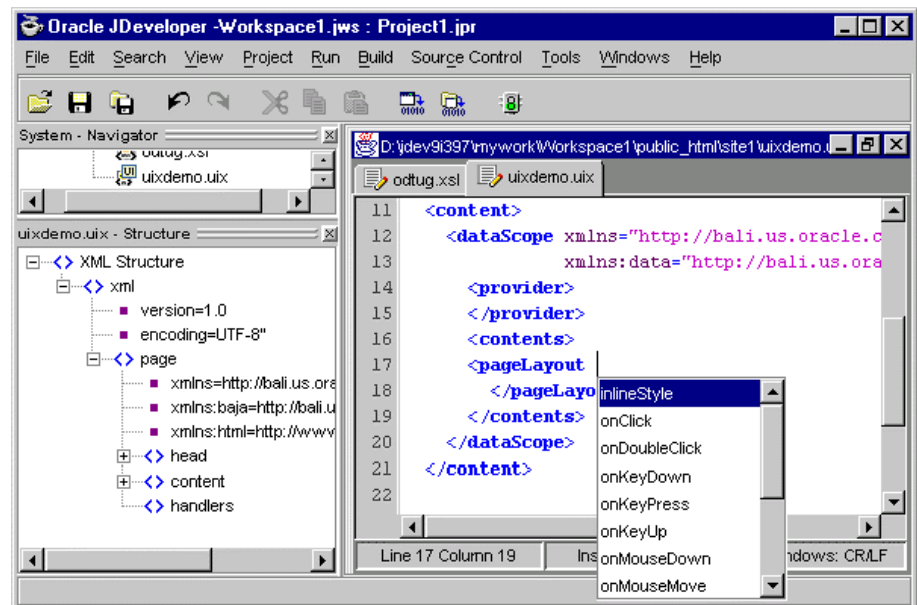


Figure 4. The XML editor with Code Insight

Other XML features include syntax highlighting, syntax checking, a tree-based structure viewer, and manipulation through the property inspector.

## **SQL and PL/SQL Tools**

JDeveloper Release 9i provides expanded features for managing connections and browsing database objects, in addition to a new PL/SQL editor and SQL viewer.

### **Managing Connections**

JDeveloper has the concept of a reusable connection, which persists the details of a database connection, including username, password, port number, hostname, and database identifier. The connection can be reused at several points in the development cycle, as you:

- Browse through existing database objects
- Edit and compile PL/SQL code (procedures, functions, packages)
- Execute and tune SQL statements
- Access data from a BC4J, JSP, DAC, and/or XSQL application
- Deploy Java code to the server

### **Browsing Database Objects**

Connections in the project navigator offer a view on several types of schemas objects in the database:

- Tables and views
- Sequences
- Synonyms
- PL/SQL code and deployed Java classes
- Oracle objects

JDeveloper has a specialized viewer for each of these object types, which displays details of the selected object. For example, the table viewer shows information about the columns in the table.

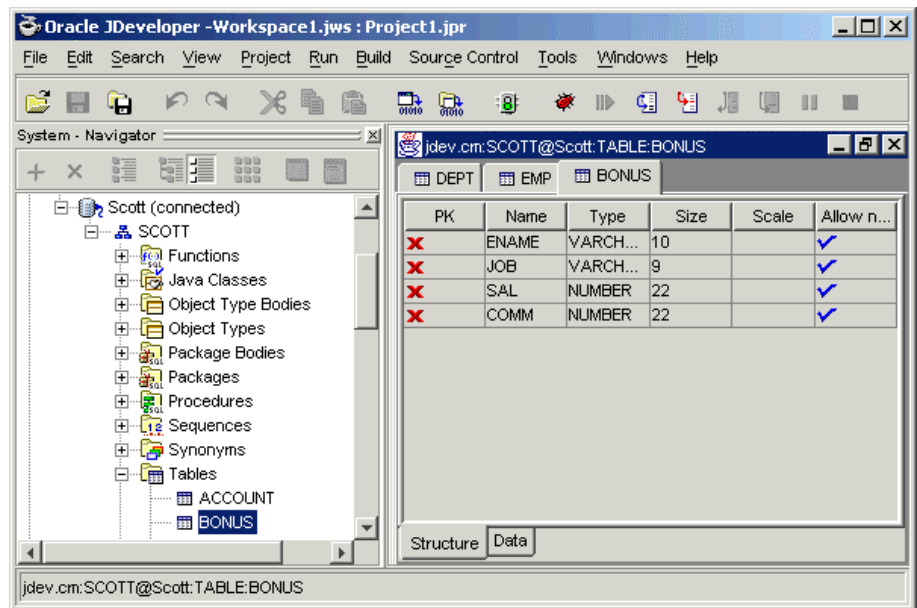


Figure 5. Browsing database objects

#### The PL/SQL Editor

The PL/SQL editor allows for editing PL/SQL objects, such as procedures, functions, and packages. With the PL/SQL editor, developers work directly on the source code from the database. To compile the PL/SQL code, JDeveloper simply sends the new source code to the database, and lets the server take care of compilation. When errors occur, they are displayed in JDeveloper.

#### The SQL Viewer

The SQL viewer provides a window where SQL statements can be executed. Using the SQL viewer is similar to using a SQL client program such as SQL\*Plus or SQLPlus Worksheet. In addition to displaying the results for a given SQL statement, the SQL viewer can also display the execution plan.

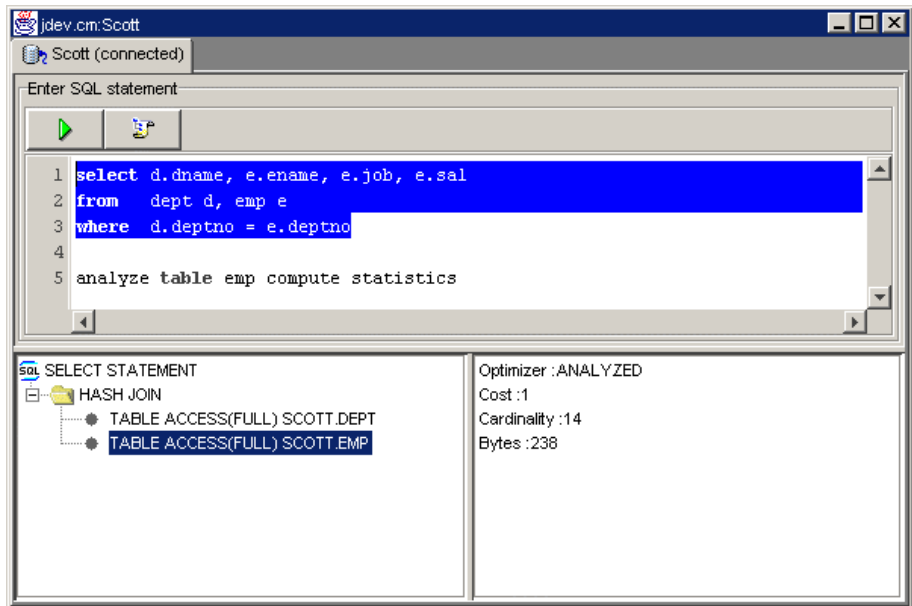


Figure 6. The SQL Viewer

## UML MODELING IS BECOMING MAINSTREAM

As the Unified Modeling Language (UML) becomes the de facto standard for software development modeling, JDeveloper will incorporate full support. The delivery of UML features in JDeveloper will be staged, introducing two UML modelers in the 9i release: the Class Modeler and the Activity Modeler.

Oracle JDeveloper makes UML modeling easily accessible to Java developers.

The overall focus of UML modeling in JDeveloper is to support the development of e-business applications, exploiting the full potential of the Oracle9i Application Server and Oracle9i Database.

The Class Modeler supports modeling and integrated code generation for Oracle's Business Components for Java (BC4J) and for native Java classes, with tight integration between the model and the code.

The Activity Modeler is specifically focused on integrating e-business applications. The generation targets are Oracle Advanced Queuing (AQ) and Oracle Workflow with JMS access and XML message payloads.

The JDeveloper modelers support standard UML usage, to which JDeveloper-specific features have been added. They also support XMI for exchange with other UML tools.

## Class Modeler

### Java Classes

Developers can use the Class Modeler to visualize and generate Java classes. The two-way generation feature automatically synchronizes code and model regardless of whether the changes are made in the code editor or the Class Modeler. The reverse-engineering facility allows developers to build models from existing code.

### Business Components for Java

Developers can also use the Class Modeler to visualize the persistent business components that make up their e-business application. From this model, JDeveloper Release 9i automatically generates the underlying implementation using the BC4J framework. The JDeveloper Class Modeler enables developers to generate fully functional BC4J applications rapidly, visually, and without programming – from UML models.

The Class Modeler implements a UML profile for BC4J that provides developers with a visual programming environment for defining the components of BC4J applications. Such components include entity objects, which represent the persistent business objects in the problem domain.

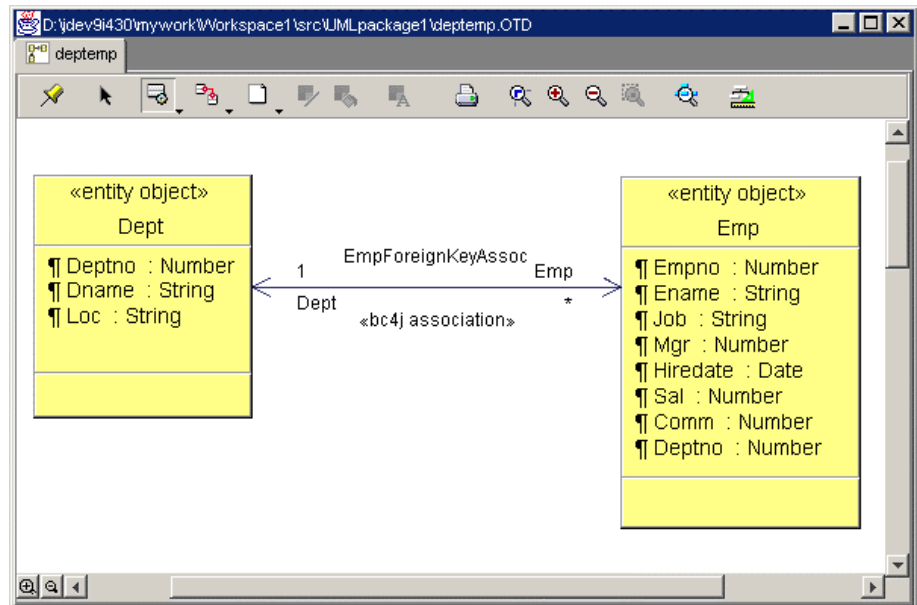


Figure 7. The Class Modeler

### Activity Modeler

The Activity Modeler is used to model business processes for e-business integration. Developers use this modeler to visualize and generate the integration between e-business applications either within a corporate intranet or across the public Internet.

The Activity Modeler allows developers to model business process flows and to capture the additional semantic data to define executable processes, routing, and queues. The process is expressed in terms of activities, subsystems (that perform the activities within partitions), and routing.

### IMPROVING CODE QUALITY

JDeveloper offers various powerful tools for improving the quality of Java code by pinpointing potential problem areas such as performance problems, memory leaks, and deadlocks.

## Execution Sampling Profiler

The Execution Sampling Profiler is used to locate performance problems by determining which methods in the code take up the most time during execution. The profiler takes a sample at regular intervals and records where the current thread is at the time of the snapshot. The samples are computed and displayed at the end of the profiling session.

Oracle JDeveloper offers powerful tools to improve the quality of Java code.

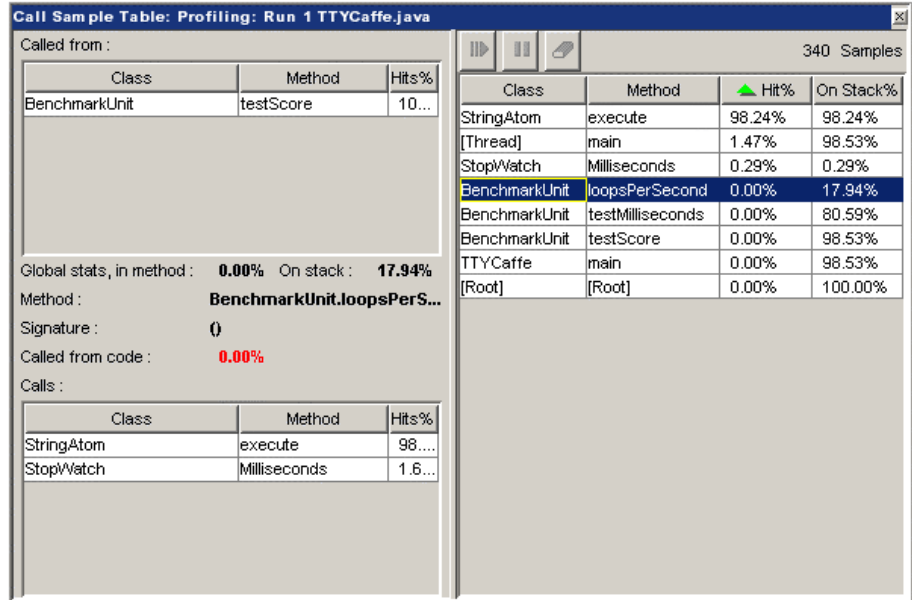


Figure 8. The Execution Sampling Profiler

Developers can interact with the information in the profiler window by drilling down into areas where performance issues are suspected. Information is available about how methods are invoked and how often. Using this tool it is easy to pinpoint where the application is spending the bulk of its time. The next task is to find out how to improve the code.

## CodeCoach

CodeCoach is a tool that provides advice on the quality and performance of Java programs. With the profiler a user may have located a problem area in the code. Running CodeCoach subsequently on this code can provide that user with many practical tips for improving the code. The tool can even fix simple problems automatically.

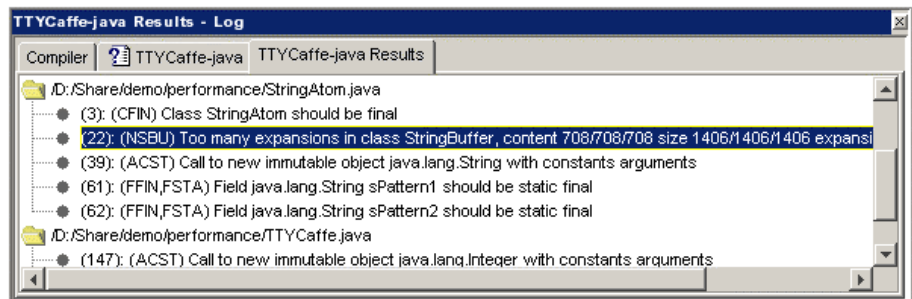


Figure 9. CodeCoach found that too many expansions were made for StringBuffer

### **Memory Profiler**

The Memory Profiler is a powerful tool for tracking down memory leaks. It takes samples at specified intervals and shows the user how and where memory is allocated and freed throughout the duration of the program.

For complex memory issues, a combination of the Memory Profiler and debugger tools such as the Heap window and the Loaded Classes window, can pinpoint the exact source of the leak.

### **Event Profiler**

The event profiler is very useful when an application consists of several layers. At a higher level than the Execution Sampling Profiler, this profiler tracks the length of specific events, no matter where they are executed. For example, a JSP application can request data from a BC4J component, which in turn issues a query to the database. The Event Profiler can easily summarize how much time was spent executing this request and how it compares to other events in the application.

### **Debugger**

Developers frequently use multiple threads to handle traffic in a Java program, introducing the chance of a deadlock. The debugger has many new features, including several new breakpoint types and better support for detecting deadlocks. The Monitors window will detect deadlocks between threads, and display the object that causes the block.

Note: The profilers, the CodeCoach tool and some of the features of the debugger only work on Windows (NT, 2000 and XP). You can however use JDeveloper on Linux or Solaris and remotely debug or profile an application that is running on a Windows machine.

## **VERSION CONTROL AND HOSTED SERVICES**

In release 3.2, JDeveloper introduced version control with the Oracle Repository. In release 9i, the Repository will be integrated more tightly into the development environment. Furthermore, JDeveloper will start supporting other source control systems, as well as WebDAV. WebDAV is a standard protocol for reading and writing files over the web and is a very promising new development in the direction of hosted development.

## **DEVELOPING JSP APPLICATIONS**

JDeveloper 3.2 provides wizards and custom tags to build JSP applications with BC4J. These custom tags provide a highly productive and flexible way to develop web applications.

JDeveloper Release 9i significantly extends the functionality in this area with the following:

- Generation of Oracle Browser Look and Feel (BLAF) pages
- JSP Code Insight
- Customizable component palette for JSP tags
- JSP tag library creation wizard
- Improved BC4J data tag library
- JSP page and application generation wizards
- Web Archive (WAR) deployment
- One-click deployment to the Oracle9i Application Server

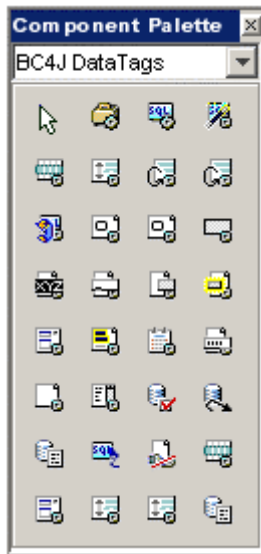


Figure 10. The Component Palette displaying JSP tags

## SUMMARY

Release 9i is the first 100% Java version of Oracle JDeveloper and is supported on Windows NT, 2000 and XP, Linux, and Solaris.

JDeveloper Release 9i covers the complete application development life cycle, from editing, compiling, testing, and tuning, to deploying and debugging. This release expands its existing version control features and introduces UML modelers.

To build complete e-business applications, JDeveloper offers support for a wide range of programming languages, including Java, XML, HTML, SQL, and PL/SQL.

To learn more about Oracle JDeveloper, or to obtain a free copy, please log on to <http://otn.oracle.com/products/jdev>.



Oracle JDeveloper Release 9i Sneak Preview  
May 2001

Oracle Corporation  
World Headquarters  
500 Oracle Parkway  
Redwood Shores, CA 94065  
U.S.A.

Worldwide Inquiries:  
Phone: +1.650.506.7000  
Fax: +1.650.506.7200  
[www.oracle.com](http://www.oracle.com)

Oracle Corporation provides the software  
that powers the internet.

Oracle is a registered trademark of Oracle Corporation. Various  
product and service names referenced herein may be trademarks  
of Oracle Corporation. All other product and service names  
mentioned may be trademarks of their respective owners.

Copyright © 2000 Oracle Corporation  
All rights reserved.