



JBUILDER 2008

Reasons to Buy

1. Leverage the latest Eclipse open source framework

The JBuilder 2008 product line builds on the common Java and Java EE development capabilities found in Eclipse 3.3 and Web Tools Platform 2.0 to meet the needs of Java developers at every level. And, because JBuilder 2008 is based on the world's most popular open source IDE framework, developers are more than likely already familiar with many of the basic features and functions found in the IDE which means a lower learning curve and shorter time to being fully productive.

In addition to the core Eclipse functionality, CodeGear also provides over one hundred third-party plug-ins and enhancements for a more complete user experience. As a trusted turn-key solution, JBuilder 2008 gives developers the confidence to expand these capabilities by providing a one-stop shop for downloading and integrating these and other technologies into a single, managed, and supported development environment.

2. Speed development with support for the latest Java technologies and application server platforms

JBuilder 2008 supports the latest commercial and open source Java application servers including Borland Enterprise Server, BEA WebLogic Application Server, IBM WebSphere, JBoss, Oracle Application Server, Oracle Containers for Java (OC4J), Apache Geronimo, Apache Tomcat, and Sun Glassfish. This means that regardless of which application server(s) you are targeting for deployment, you can use a single IDE for development.

The graphical EJB Workbench in JBuilder 2008 provides a simplified RAD development experience for both novice and experienced Java EE developers alike. Legacy projects using EJB 2.x are easily converted to the new annotation-based EJB 3.x specification. Easily create EJBs and model relationships, security and OCL constraints. JBuilder's patented LiveSource technology insures that changes to the source code, annotations, properties view and graphical view are always in sync.

3. Easily move JBuilder projects forward

JBuilder 2008 advances developers to the latest Eclipse-based JBuilder platform with backward compatibility support for previous versions of JBuilder. Easily import existing projects into the new JBuilder 2008 IDE. Enjoy the flexibility and cost-efficiencies associated with the Eclipse open source framework, and the enterprise-class functionality, reliability, and support you require.

4. Experience unparalleled productivity and code reuse

The Application Factories functionality in JBuilder 2008 introduces an application-driven development paradigm, where the structure, evolution, and logic behind the development of the application are checked into version control along with the source code for the application itself. The Application Factory records the developer's intent as an application is being created so that the workings of the code can be easily ascertained at a later date. And while Application Factories will certainly yield higher levels productivity during the development of an application, one of the biggest benefits may come when that application has to be modified or enhanced. This is because the metadata, which stays attached to the code when it is created, can be opened by any subsequent developer to understand the context and purpose of code snippets, methods, and classes which are part of the application.

5. Easily create sophisticated Swing-based applications

Swing Designer in JBuilder 2008 is a comprehensive set of user interface construction tools designed to enable developers to rapidly create Java Swing applications. The visual designer, wizards, editors, and intelligent

layout assistance supports Swing/AWT development including the most popular controls and layout managers. Swing Designer provides full bi-directional code generation, with the visual design and the source always in 100% synchronization. JBuilder developers can focus on creating application specific functionality rather than coding the low-level logic required for graphical user interfaces.

6. Analyze, debug, and tune applications within the IDE

High-level performance-related data displayed in real time allows developers to understand whether a performance issue is related to CPU, memory, or both. Automatic Memory Leak Detector monitors the evolution of memory usage over time for the immediate identification of potential memory leaks. Real-time monitoring of object allocations to understand how the profiled program uses the virtual machine memory; Allocation Backtrace View allows developers to identify the code or part of the program responsible for instance allocations. Thread Debugger gives real-time display of the progress of all threads running within the Virtual Machine. Code Coverage provides a Real-time Class Coverage View to quickly see the coverage for each class and identify classes not fully covered.

7. Increase individual and team productivity

JBuilder 2008 provides team and project manageability through the TeamInsight collaboration feature. Individual developers have a unified real-time view of their project responsibilities for bugs, change requests, code notes, tasks and requirements, and the entire team has a shared project Web portal with live data and statistics on team vector and velocity. ProjectAssist provides simple, single point installation and configuration of a complete developer tool stack for bug tracking, source code management, project planning/tracking, requirements management and continuous integration builds, saving days or weeks of manual configuration and integration. JBuilder 2008 also provides integrated support for live and off-line editing of all project artifacts including defects, source, project tasks and requirements. All projects are automatically synchronized when on-line.

8. Accelerate creation of Web services

The Web Services Workbench in JBuilder 2008 is a visual, two-way designer for creating Axis based Web service-enabled applications. The workbench supports instant navigation between Web services components displayed in the designer (e.g. WSDL, methods, operations) and the corresponding source code. Key features of the Web Services Workbench include the ability to:

- Import a WSDL document to build a test client for a Web service discovered through a UDDI registry
- Build a SOAP server to act as the base for a Web service
- Expose a simple Java class as a Web service
- Expose an EJB Session Bean as a Web service
- Debug a Web service and a Web service client concurrently
- Dynamically generates and deploy Web service components during project builds

9. UML modeling and code archeology

JBuilder 2008 provides full Java UML 2.0 modeling capabilities, including support for all standard modeling diagrams such as Sequence diagrams, Use Case diagrams, Activity diagrams, Class diagrams, Component diagrams, Deployment diagrams, State Machine diagrams, Composite Structure diagrams, Communication diagrams, and Web Services diagrams. Both HTML format and templated document generation is also supported. The LiveSource technology in JBuilder 2008 simultaneously synchronizes changes made to the UML models in the code, and vice versa.

10. Audits and Metrics

CODE GEAR FROM Borland

JBUILDER

Reasons to Buy

The audits and metrics capabilities in JBuilder 2008 can be used to solve the code quality, code review and code dependency issues typically faced in software development. JBuilder's audits and metrics are designed to aid developers in finding and fixing problems in their code earlier in the development process. JBuilder's audits perform a static analysis of your source code, identifying *coding* problems like potential race conditions, unchecked exceptions and more. JBuilder's metrics provide a similar analysis for *design* problems in your project.