

# Borland® JBuilder® 2005

## Feature Matrix

FOUNDATION      DEVELOPER      ENTERPRISE

### Customizable integrated development environment

#### AppBrowser™ IDE environment overview

Professional IDE with integrated AppBrowser,™ editor, debugger, compiler, and help	✓	✓	✓
Cross-platform development, with customizable look-and-feel, on Windows,® Linux,® Mac OS® X, and Solaris™ operating systems	✓	✓	✓
JBuilder® IDE hosted on Java™ 2 SDK 1.4 for enhanced platform interoperability and performance	✓	✓	✓
Includes Java 2 SDK 1.4 with built-in HotSpot™ Client Virtual Machine for increased performance	✓	✓	✓
True two-way visual IDE with Two-Way-Tools™	✓	✓	✓
Build Java™ applications, applets, JavaBeans,™ Enterprise JavaBeans™ (EJB™) servlets, and JavaServer™ Pages (JSP™) with no proprietary code or markers	✓	✓	✓
Unicode® -enabled editor with mapping to single- and double-byte code pages	✓	✓	✓
Integrated HTML and Javadoc™ viewer with adjustable font sizes	✓	✓	✓
Javadoc browser dynamically generates Javadoc from Javadoc embedded in source code	✓	✓	✓
Class browser for viewing class structures	✓	✓	✓
Integrated Unified Modeling Language™ (UML®) browser for code visualization			✓
Java help system with full Boolean text search for easier access to documentation	✓	✓	✓
Embedded help viewer in content pane	✓	✓	✓
Built-in API browser for class file methods and data members	✓	✓	✓
StructureInsight™ for rapid file navigation through source files	✓	✓	✓
OpenTools API for adding new and customizing existing functionality of the JBuilder environment	✓	✓	✓
Documentation print capability inside the AppBrowser	✓	✓	✓
Customizable support for print to printer or to an HTML file for creating superior project documentation	✓	✓	✓
QualityCentral Client to log issues with Borland support	✓	✓	✓

#### Supported standards

Support for the latest Java standards, including Java 2, Java 2 JFC/Swing, XML, Java2D, Message Queue, Java collections, Accessibility API, Speech API, Java™ 2 Platform, Enterprise Edition (J2EE™) 1.3, EJB 2.0, and more	✓	✓	✓
Built-in support for the latest Java 2 platform	✓	✓	✓
Develop, debug, and compile with Java 2 SDK 1.3, 1.4, and 5.0	✓	✓	✓

# Borland®

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Support all JDK™ 5.0 (1.5) language features, including generics	✓	✓	✓
Supports all JDK 1.4 language features, including asserts	✓	✓	✓

### User interface

User-friendly and intuitive IDE interface with consistent behavior and navigation	✓	✓	✓
Newly improved windowing system	✓	✓	✓
Dockable message/output panes with enhanced functionality	✓	✓	✓
Dockable project and structure panes	✓	✓	✓
Dockable file browser	✓	✓	✓
File status decorations in project pane	✓	✓	✓
Minimize message, project, file browser, and structure pane as iconified tabs in the IDE border	✓	✓	✓
Open any file in a new window by dragging the file icon out of the project pane or file browser and dropping into the content pane or onto the desktop	✓	✓	✓
Structure Pane Visibility filters to show type of class, return types, and field types, group by visibility, group by type and inheritance, and sort alphabetically	✓	✓	✓
Drag and drop file open	✓	✓	✓
File properties tool tip for files in content pane tree	✓	✓	✓
Project pane with file drag and drop enabled	✓	✓	✓
Configurable IDE Personality for each project to streamline the user interface	✓	✓	✓
Right-click in the project pane to add classes, interfaces, packages, JSP, or any files	✓	✓	✓
Right click in the structure pane to add new field, method, constructor, Javadoc, or inner class	✓	✓	✓
AppBrowser for integrated management of projects, classes, and source	✓	✓	✓
Project-level class filter configuration			✓
Directory view project node with project-aware file support		✓	✓
Object gallery with access to wizards with consistent behavior and navigation across all wizards	✓	✓	✓
Themed audio feedback with customizable audio themes	✓	✓	✓
Message view toggler to turn message view on or off	✓	✓	✓
Enhanced status bar	✓	✓	✓
Single-click message or editor tab closing	✓	✓	✓
Add favorites to open file dialog	✓	✓	✓
Project folders and resource management for increased productivity		✓	✓
Package view for displaying source packages in project view with customizable package nesting levels	✓	✓	✓
To-do tag support and project level access to all to-do tags	✓	✓	✓

### Section 508 accessibility

Access JBuilder functionality through keyboard	✓	✓	✓
Swing accessibility framework usage	✓	✓	✓
Integrate OS system font size as selected	✓	✓	✓
All color coding complemented by non-color-based functions	✓	✓	✓
Navigation across panes and tabs available from keyboard	✓	✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
High-contrast color scheme	✓	✓	✓
Editor text zoom	✓	✓	✓
Message severity status with multiple communications schemes (color, icon, sound, and symbol)	✓	✓	✓
Browser splitter access via keyboard	✓	✓	✓

### JBuilder installation support

Product uninstall	✓	✓	✓
Dynamic product license registration	✓	✓	✓
Product install from server	✓	✓	✓
Generic product installation (alternate setup) available for unsupported platforms			✓
Automatically import settings from older version of JBuilder during install or using a wizard	✓	✓	✓
Concurrent Version System (CVS) client installation	✓	✓	✓
Crystal Reports® for Borland® JBuilder® <sup>1</sup>			✓
eBay® SDK for Java™ <sup>1</sup>			✓
Vignette® Portal plug-in for Borland® JBuilder® <sup>1</sup>			✓
Sybase® EAServer™ integration module for Borland® JBuilder® <sup>1</sup>			✓
Fortify Software for Borland® JBuilder®			✓

### UML code visualization

UML class and package diagram representation of code: Limited Class Association Diagram, Reverse Class Association Diagram, Limited Class Dependency Diagram, Reverse Class Dependency Diagram, and Class Inheritance Diagram			✓
View multiple classes, including inner classes and anonymous classes, through a tabbed user interface			✓
Full browser-based navigation functionality to navigate diagrams and change focus to new classes and packages			✓
Filter to exclude chosen classes packages from diagrams for a project			✓
Choose to include libraries on a project basis to view dependencies in diagrams			✓
Structure pane tree view of package or class with expandable folders; navigate to diagrams from tree view			✓
Customize fonts and colors in diagram presentation			✓
Fast access to refactoring functions directly from diagrams: rename, move, find reference, find definition, change parameters, extract interface, and introduce super class			✓
Limited package dependency diagram			✓
Print diagram, or save diagram as an image			✓
Quick navigation from diagram to source code and documentation			✓
Diagram annotation to show number of subclasses			✓

### Sample applications

Generic sample applications	✓	✓	✓
OpenTools sample applications showing diff-view and keymap definitions implementation	✓	✓	✓

<sup>1</sup> Support for these additional software products is provided by their respective manufacturers

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Multilingual online shopping sample application with source code	✓	✓	✓
OpenTools sample applications for adding custom wizards, viewers, and many more	✓	✓	✓
Extensible version control system OpenTools sample with integrated CVS functionality with support for update, merge, add, and check-in/out		✓	✓
Shopping cart example using JSP		✓	✓
Bookstore sample application using JSF		✓	✓
XML tutorial showing examples of validation, databinding, XSL, database import/output, and more			✓
Online store real-life sample program with credit-card approval, order entry, and order tracker functionality			✓
Credit-card approval client using CORBA®/RMI			✓
Bookstore sample application to demonstrate InternetBeans™ Express with DataExpress™ for EJB and more			✓

## Project, build, and run management

### Project management

XML-based Project Manager supporting JPX project file format	✓	✓	✓
JDK switching in projects for developing to different Java JDK versions and Java 2 SDK editions (Java™ 2 Platform, Micro Edition™ (J2ME™), Java™ 2 Platform, Standard Edition (J2SE™) and J2EE	✓	✓	✓
Project wizard for conveniently and effectively starting new projects	✓	✓	✓
Project Import wizard to import existing code projects		✓	✓
Project groups		✓	✓
Add another project as a library			✓
Right-click to add any files or packages to project	✓	✓	✓
Wizard to import existing Java source packages or any other resources into current project source	✓	✓	✓

### Build management

Project build configuration management infrastructure based on Ant 1.6 supports custom enterprise build processes and external Ant 1.5 builds	✓	✓	✓
Accept build targets as arguments with the external command-line build	✓	✓	✓
Extend the build system with custom build tasks	✓	✓	✓
Customize build menu and toolbar with custom build targets	✓	✓	✓
Support for SQLJ™ (Supports the Oracle® and IBM® DB2® translators for SQLJ to generate Java from SQLJ)			✓
External build task wizard to create external tasks that allow execution of external shell or console commands as part of the build process		✓	✓
Sample to illustrate the new build system: using RetroGuard to obfuscate JAR files	✓	✓	✓
Filter selected packages from build process; includes a filter overview folder to view filtering rules		✓	✓
Package Java applications as native executables		✓	✓
Select compiler including Borland Make for Java (bmj) or javac, the Java compiler		✓	✓
Fast Java compiler for rapid compiles	✓	✓	✓
Command-line compiler, dependency checker, and build		✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Compile option to strip debug information (source, line, and member) to reduce class file size		✓	✓
Built-in local obfuscator to protect Java source code		✓	✓
JBuilder® Archive Builder™ to quickly build deployable archive types for applets, applications, JavaBeans, and OpenTools	✓	✓	✓
Fully manage archive content using either inclusion or exclusion logic	✓	✓	✓
Smart dependency checking for class and resource dependencies		✓	✓
Adding multiple classes or entire packages to deployment archive	✓	✓	✓
Archive support for native executable		✓	✓
Java archive and PkZip® (JAR and zip) format support	✓	✓	✓
Deployment of user-defined file types	✓	✓	✓
Auto-archive to update archive files at build time	✓	✓	✓
Create manifest information automatically or manually	✓	✓	✓
Archive source and documentation	✓	✓	✓

### Run management

Runtime configurations for managing multiple executables per project		✓	✓
OpenTools Runner to load and run OpenTools projects without restarting JBuilder	✓	✓	✓
Runtime configurations enabled to run in debug or optimize mode	✓	✓	✓
Configure individual runtime configurations to run with any JDK and libraries	✓	✓	✓
Run, debug, and optimize an executable JAR file within JBuilder, without specifying the main class in the runtime configuration	✓	✓	✓

### Apache™ Ant integration

Integration with external Apache™ Ant 1.6 builds	✓	✓	✓
Add Ant nodes to projects and execute Ant targets from within JBuilder	✓	✓	✓
Navigate Ant files using target nodes in project pane	✓	✓	✓
Set breakpoints in Ant files to debug Ant scripts		✓	✓
Use macros within Ant property values	✓	✓	✓
Add custom Ant-based build tasks with custom Ant libraries to run Ant from within JBuilder	✓	✓	✓
Rapid navigation from Ant build-error messages to source files	✓	✓	✓
Export project configurations as an Ant Script		✓	✓

### Two-way code migration

Create project from existing code wizard		✓	✓
Import project from version management system		✓	✓
Import Enterprise EJB applications from existing code			✓
Import Web applications from existing code		✓	✓
Import VisualCafé™ project wizard			✓

## Java coding productivity

### Code editing

Line numbering	✓	✓	✓
----------------	---	---	---

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Show white space characters	✓	✓	✓
Smart MemberInsight for specific types or typecasts appropriate for the current context	✓	✓	✓
Code folding: fold methods, imports, and class body without losing line numbering	✓	✓	✓
Unlimited line length editing	✓	✓	✓
Single keystroke selection modification	✓	✓	✓
Garbage collection manager for heap alerts and garbage collection trigger	✓	✓	✓
Scoping of method in editor margin	✓	✓	✓
Drag-and-drop text editing	✓	✓	✓
Access refactorings, find definitions, and find references from within the editor	✓	✓	✓
Advanced search-and-replace options with pattern matching	✓	✓	✓
Search combo box now available from the toolbar	✓	✓	✓
Global search and replace on all files in path	✓	✓	✓
Parallel search execution to perform multiple searches simultaneously	✓	✓	✓
Search in path using regular expressions	✓	✓	✓
Go to class member dialog to navigate to a class member in open class file	✓	✓	✓
Hyperlink navigation to any class member—symbol changes to hyperlink to navigate to symbol definition	✓	✓	✓
Emacs™ style single-line indenting and automatic code formatting	✓	✓	✓
Use regular expressions in search expressions	✓	✓	✓
Highlighted brace matching	✓	✓	✓
Automatic closing brace insertion	✓	✓	✓
Syntax highlighting for multiple file types, including Java, HTML, and C/C++	✓	✓	✓
Advanced syntax highlighting for XML, WML, IDL, JSP, and XSL, and Cascading Style Sheets (CSS)		✓	✓
Customizable smart keys with auto-indent option	✓	✓	✓
Advanced keymap editor to create new keymaps or modify existing keymaps	✓	✓	✓
Import or export keymaps	✓	✓	✓
CUA® editor keymap	✓	✓	✓
Emacs editor keymap	✓	✓	✓
Microsoft® Visual Studio® keymap	✓	✓	✓
Brief® editor keymap	✓	✓	✓
Macintosh® keymap	✓	✓	✓
CodeWarrior® for Mac® keymap	✓	✓	✓
Visual Keymap editor for customizing environment actions and key strokes definitions	✓	✓	✓
Quick access keymap drop-down to toggle keymap	✓	✓	✓
Sync Edit to simultaneously edit multiple identical identifiers in a block of code	✓	✓	✓
Right click in editor to add new field, method, constructor, Javadoc, or inner class	✓	✓	✓
Right-click on class to create, rename, or delete class fields and properties	✓	✓	✓
Add bookmarks to code and link to shortcut keys	✓	✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
<b>Code audits</b>			
Run audits on individual files		✓	✓
View audit results in structure pane and navigate to location in code by double-clicking on the warning		✓	✓
Coding style audits		✓	✓
Declaration style audits		✓	✓
Documentation audits		✓	✓
EJB-specific audits		✓	✓
Naming style audits		✓	✓
Performance audits		✓	✓
Possible error audits		✓	✓
Superfluous content audits		✓	✓
Expression audits		✓	✓
Branch and loop audits		✓	✓
Design flaw audits		✓	✓
<b>Code formatting</b>			
Smart automatic code formatting	✓	✓	✓
Set indenting	✓	✓	✓
Format tab size	✓	✓	✓
Set end-of-line characters	✓	✓	✓
Select multiline characters	✓	✓	✓
Format continuation indent	✓	✓	✓
Format braces	✓	✓	✓
Set spaces	✓	✓	✓
Select blank lines	✓	✓	✓
Format event handling	✓	✓	✓
Format import statements	✓	✓	✓
Preview before setting preferences	✓	✓	✓
Access quick format via keyboard, or menu from editor	✓	✓	✓
Import/export code formatting settings	✓	✓	✓
JBuilder® SmartDiff™ across different code formats	✓	✓	✓
<b>Code templates</b>			
Configurable Smart Code templates to automate common coding tasks	✓	✓	✓
Macros in templates	✓	✓	✓
Templates enabled with Sync Edit upon insertion	✓	✓	✓
File awareness in templates: template creation for any file types	✓	✓	✓
Right-click on code selection to generate template from code selection	✓	✓	✓
Open Tools API to enable additional file type template categories	✓	✓	✓
Class-aware templates integrated with automatic insertion of import statement	✓	✓	✓
Surround block templates to surround selection with template code	✓	✓	✓
Import/export templates to file	✓	✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
<b>CodeInsight™</b>			
CodeInsight™ to speed coding and reduce syntax errors	✓	✓	✓
Code Completion automatically displays package, members, methods, and events for Java classes and JavaBeans	✓	✓	✓
CodeInsight automatic import statement completion and optimization	✓	✓	✓
CodeInsight package content review	✓	✓	✓
Code Parameter ToolTips automatically displays procedure, method, and event parameter lists	✓	✓	✓
Customizable CodeInsight for key-binding and appearance	✓	✓	✓
Visual indication of public, protected, private, and deprecated members	✓	✓	✓
Package completion wizard to help ensure proper syntax	✓	✓	✓
Package Browser to list classes by package	✓	✓	✓
<b>ErrorInsight™ and Forward Code Assist</b>			
ErrorInsight™ to automatically display common coding errors	✓	✓	✓
Red underlining of syntax errors	✓	✓	✓
Gray-out unused imports, methods, and variables	✓	✓	✓
Add method wizard when error fix requires creation of new method	✓	✓	✓
Add throw cause and surround with try/catch to fix uncaught exceptions	✓	✓	✓
Error prompt to fix "type not found" errors	✓	✓	✓
Error prompt to fix "methods not found" errors	✓	✓	✓
Error prompt to fix "constructor not found" error	✓	✓	✓
Error prompt to fix "variable not found" errors	✓	✓	✓
Error prompt to fix class name or filename errors	✓	✓	✓
Error prompt to fix packages or directories	✓	✓	✓
Error prompt to fix type mismatches	✓	✓	✓
Error prompt to fix class declarations	✓	✓	✓
Fix missing import packages error	✓	✓	✓
ToolTips to correct missing imports	✓	✓	✓
Error prompt to fix missing import classes errors	✓	✓	✓
<b>Debugging</b>			
Debugging of multiple runtimes per project		✓	✓
Debugger with breakpoints, stepping, and watches for detecting bugs quickly	✓	✓	✓
Built-in custom view for Map and Collection objects	✓	✓	✓
Quick Step Into to step directly into the first parameter in the method signature	✓	✓	✓
Support for the HotSpot Serviceability Agent to examine the heap as well as access thread stacks	✓	✓	✓
Debugging of Java 2 applications based on the industry-standard Java™ Platform Debugger Architecture (JPDA)	✓	✓	✓
Local variables view for all stack frames for all threads	✓	✓	✓
Runtime evaluation for class members, including variables and constants	✓	✓	✓
Advanced debugger features, including conditional breakpoints, breakpoint logging, evaluator, step out, and run to end of method		✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Debugging in multiple Java 2 SDK versions and editions	✓	✓	✓
Advanced runtime evaluation and dynamic invocation of methods		✓	✓
Evaluator variable inline assignment		✓	✓
Multiple debug configurations for managing multiple debug sessions per project		✓	✓
Simultaneous display of thread and stack-frame context		✓	✓
Floating debug windows to view multiple debug views simultaneously		✓	✓
Set Execution Point function to move execution point while debugging (JDK 1.4 HotSwap™ support)		✓	✓
Smart Swap to swap out classes while debugging (JDK 1.4 HotSwap support)		✓	✓
Add non-Java program language source line breakpoint (JSR 45 support)		✓	✓
Set source view to toggle between non-Java code and generated Java source while debugging		✓	✓
Simultaneous multiprocess debugging to simplify development of partitioned applications	✓	✓	✓
Debugging of JDK with Java 2 JPDA debug API support		✓	✓
Line, class, method, and exception breakpoints to narrow down bugs more efficiently		✓	✓
SmartStep for configuring debug stepping (step into, over, out) behavior in class source	✓	✓	✓
ToolTip Evaluation for on-the-spot display of member instances		✓	✓
JBuilder® ExpressionInsight™ with detailed structure view of member instances		✓	✓
Breakpoint actions for programmable control on break events		✓	✓
Breakpoint expressions for getting an automated view on developer logging information		✓	✓
Sourceless debugging for stepping into classes without associated source code		✓	✓
Exception source trace to find source location of exception incident		✓	✓
Servlet debugging, breakpoints, evaluators, watches, etc.		✓	✓
JSP source code debugging with Java debugging support		✓	✓
Other non-Java source code debugging, including SQLJ and LegacyJ®			✓
Thread deadlock, stalls, and race conditions detection on precise error location		✓	✓
Keep-thread-suspended option for tracking thread race conditions		✓	✓
Cross-process breakpoints to control stepping behavior for debugging server code		✓	✓
Attach to and debug running process		✓	✓
Multiplatform and remote debugging for debugging complex distributed applications on a variety of platforms		✓	✓

### Refactoring

Invoke refactoring from structure pane and class browser	✓	✓	✓
Distributed refactoring to distribute refactorings among various projects	✓	✓	✓
Automatic update of Runtimes and archive nodes during refactorings	✓	✓	✓
Refactorings into non-Java files such as deployment descriptors and JSP files		✓	✓
JDK 5.0 refactoring to convert existing loops to JDK 5.0 enhanced loops	✓	✓	✓
JDK 5.0 refactoring for autoboxing and auto-unboxing to easily convert	✓	✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
between primitive types and their Object-based counterparts.			
JDK 5.0 refactoring for generics	✓	✓	✓
Extract a method: Convert a code selection to a method	✓	✓	✓
Introduce a variable: Select a complex expression and convert to a variable	✓	✓	✓
Introduce a field: select a complex expression and convert to a field	✓	✓	✓
Change a methods signature: add, remove, or reorder parameters	✓	✓	✓
Remove used parameters in method signature; replace with local variable	✓	✓	✓
Surround with try/catch: Surround a code selection with try/catch and automatically add specific catch blocks for all detected and selected exceptions	✓	✓	✓
Rename packages and update all references	✓	✓	✓
Rename classes and update all references	✓	✓	✓
Rename methods and update all references	✓	✓	✓
Rename fields, method arguments, and local variables; update all references	✓	✓	✓
Move classes and update all references	✓	✓	✓
Pull up method into a superclass of current class	✓	✓	✓
Push down method into a subclass of current class	✓	✓	✓
Pull up field into a superclass of current class	✓	✓	✓
Push down field into a subclass of current class	✓	✓	✓
Introduce a superclass from multiple classes	✓	✓	✓
Extract Interface	✓	✓	✓
Delegate to instance: Create delegation methods for an instance variable in class	✓	✓	✓
Search and find all references	✓	✓	✓
Find definitions for a symbol	✓	✓	✓
Validate new names based on context	✓	✓	✓
Refactoring cancel, undo, and redo	✓	✓	✓
Check for limitations before refactoring	✓	✓	✓
Refactor references not referenced in compiler, including references in Javadoc entries	✓	✓	✓
Invoke refactoring from menu and code editor	✓	✓	✓
Invoke refactoring from UML diagrams			✓
<b>Code generation</b>			
Implement interface wizard to assist in implementing Java interfaces (supports inner classes)	✓	✓	✓
Method wizard to implement override or abstract methods (supports inner classes)	✓	✓	✓
Class wizard to simplify the creation and definition of classes; package, base class, and class options	✓	✓	✓
JavaBean wizard to radically simplify coding and speed implementation of JavaBeans	✓	✓	✓
<b>Javadoc support and JavadocInsight</b>			
Integrated Javadoc syntax checker provides interactive feedback for creating compliant Javadoc entries	✓	✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Javadoc wizard to generate documentation node in project and automatically create Javadoc formatted according to doclets. Select which packages are documented and format of output		✓	✓
Automate Javadoc generation as part of the build process		✓	✓
Interactive Javadoc syntax checker	✓	✓	✓
Doclet support to govern output format: includes standard JDK 1.3 doclet, JDK 1.4 doclet, and JDK 1.1 doclet		✓	✓
Javadoc Insight to speed coding of Javadoc tags and reduce syntax errors	✓	✓	✓
Add custom Javadoc tags to JavadocInsight		✓	✓
Color highlighting of Javadoc syntax	✓	✓	✓
Right-click on Javadoc conflicts in structure pane to automatically resolve conflicts	✓	✓	✓

### Internationalization support

Resource string wizard to eliminate hard-coded strings and ease creation of resource bundles		✓	✓
Convenient visual editing of individual property files		✓	✓
Simultaneous editing for multiple language versions of resource bundles		✓	✓
Support for language translations within the visual designer		✓	✓

### JBuilder® BeansExpress™ for rapidly creating Java components

JavaBeans included	200+	300+	300+
JBuilder® BeansExpress™ to rapidly create reusable JavaBeans		✓	✓
Creation of property getter/setter methods from existing members		✓	✓
Synchronized scrolling between BeansExpress and implementation code		✓	✓
Source code for JavaBeans			✓
Visual bean designers to easily create properties, events, and BeanInfo		✓	✓
JBuilder® BeanInsight™ to create and manipulate JavaBeans properties and validate JavaBeans		✓	✓
Iterative design of entity beans for EJB			✓

## Web development

### Supported platforms

Borland® Enterprise Server, AppServer™ Edition 6.0, Release Pack 1 support		✓	✓
Borland® Enterprise Server 5.2.1 support			✓
Tomcat 4.1.30 with Servlet 2.3 and JSP 1.2		✓	✓
Tomcat 5.0.27 with Servlet 2.4 and JSP 2.0		✓	✓
BEA® WebLogic® 7.0 SP5 servlet engine support			✓
BEA® WebLogic® 8.1 SP3 servlet engine support			✓
IBM® WebSphere® 4.0.7 servlet engine support			✓
IBM® WebSphere® 5.0.2.4 servlet engine support			✓
IBM® WebSphere® 5.1.0.4 servlet engine support			
Sybase EAServer 4.1, 4.2, 5.0, 5.1 servlet engine support			✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
<b>MVC-based Struts designer</b>			
Design surface to program Struts applications			
Two-way support for form beans (including DynaBeans™) and actions		✓	✓
Two-way support for tiles		✓	✓
Tight integration between tiles and struts		✓	✓
Two-way drag and drop flow design		✓	✓
ErrorInsight in designer for missing or unimplemented struts artifacts		✓	✓
Support for validation		✓	✓
HTML/JSP to Apache Struts converter wizard		✓	✓
Struts action form creation wizard		✓	✓
Struts action creation wizard		✓	✓
Generate JSP input form from Struts action form		✓	✓
Struts visual configuration editor		✓	✓
Web application wizard with Struts support		✓	✓
Struts framework support in JSP wizard		✓	✓
Struts 1.1 support		✓	✓
<b>JavaServer™ Faces (JSF) Designer</b>			
Support for JSF 1.1 Sun Reference Implementation		✓	✓
Auto configuration of Web applications for JSF		✓	✓
Visual JSF flow designer to edit navigation rules and application flow		✓	✓
JSF view editor to edit JSP-based views		✓	✓
JSF configuration editor with visual rules editor		✓	✓
Customizable drag-and-drop JSF component palette		✓	✓
Searchable component palette		✓	✓
ErrorInsight and TagInsight for JSF tags and components		✓	✓
Smart generation of backing beans		✓	✓
<b>Web deployment</b>			
Advanced error detection of embedded Java source in JSP, by precompiling before deploying to servlet engine		✓	✓
Web Start launcher wizard		✓	✓
WAR Web application archives		✓	✓
Visual forms-based Two-way Deployment Descriptor Editor according to Servlet 2.2/2.3 specification		✓	✓
Deployment Descriptor CodeInsight via XML TagInsight		✓	✓
<b>Java Web Start support</b>			
Web Start applet and application packaging support		✓	✓
Java Web Start-ready applications for Java™ Network Launching Protocol (JNLP) creation		✓	✓
JNLP support v1.0.0		✓	✓
Web Start homepage generation with Java plug-in detection		✓	✓
Web Start wizard creates Java Network Launching Protocol (JNLP) and generates home page with Java plug-in detection		✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
<b>Integrated Web development</b>			
HTML4, XML, and CSS2 cascading style sheet layout viewer		✓	✓
Web Modules to support web application configuration management		✓	✓
Create Web modules from directories		✓	✓
TCP Monitor for debugging web applications		✓	✓
HTML and JSP TagInsight		✓	✓
TagInsight inspector for HTML and JSP		✓	✓
Tag palette for JSP and HTML for drag and drop tag editing		✓	✓
Documentation Type Definition (DTD) support and extensible dynamic tags handling		✓	✓
Servlet wizard to create HTML, XHTML, XML, and WML Java servlets based on Servlet 2.2 or 2.3		✓	✓
JSP wizard to rapidly create JSP (JavaServer Pages) Web applications based on JSP 1.1 or 1.2		✓	✓
JSP/Servlet framework support		✓	✓
Struts 1.0 framework		✓	✓
InternetBeans Express 1.1 framework		✓	✓
JavaServer Pages™ Standard Tag Library(JSTL) framework 1.0 and 1.1		✓	✓
JSF 1.1		✓	✓
JSR-45 based JSP debugging support with Tomcat 5.0		✓	✓
<b>Cocoon framework</b>			
JavaScript™ execution and viewing for most common JavaScript tasks			✓
Support for Servlet 2.2/JSP 1.1 and Servlet 2.3/JSP 1.2 standards			✓
Plug-in-ready servlet engine support			✓
Option to switch between servlet plug-ins			✓
Tomcat 3.1, 3.2, 4.0, and 4.1 engine plug-in			✓
CSS templates for creating presentation layers for XML data			✓
JBuilder® StructureInsight for XML, JSP, HTML, WML, and XSL source files			✓
<b>Advanced Web development for Java servlet support</b>			
Rapid development of dynamic servlet based thin-client Java services		✓	✓
Servlet 2.3 filters and listeners		✓	✓
Includes servlet templates		✓	✓
Servlet wizard for creating thin-client HTML, XHTML, XML, and WML Java servlets based on Servlet 2.2, 2.3, and 2.4		✓	✓
Local execution of servlet		✓	✓
Local debugging of servlet		✓	✓
Remote execution of servlet		✓	✓
Remote debugging of servlet		✓	✓
<b>Advanced Web development for JSP support</b>			
JSP tag library support		✓	✓
JSP tag library-aware CodeInsight		✓	✓
Tag library integration with JSP/Servlet framework		✓	✓
Rapid development of dynamic JSP-based thin-client Java services		✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
JSP wizard for rapidly creating JSP Web applications based on JSP 1.1, 1.2, 2.0, and tag libraries		✓	✓
Executes JSP in the built-in Web server		✓	✓
JSP syntax highlighting and structure view		✓	✓
Includes servlet and JSP templates		✓	✓
Local execution of JSP		✓	✓
Easier launch of JSP (WebRun and WebDebug)		✓	✓
Remote execution of JSP		✓	✓
Local inline debugging of JSP		✓	✓
Native JSP debugging, with full breakpoint, watches, evaluation, and context information support		✓	✓
CodeInsight, ErrorInsight, ToolTip Evaluation for JSP-embedded Java		✓	✓
JSR-45 based JSP debugging support with Tomcat 5.0		✓	✓

### InternetBeans Express for component-based, data-driven applications

InternetBeans Express presentation components for rapidly creating Web-driven Internet applications based on DataExpress	✓	✓
InternetBeans Express JSP/Servlet framework	✓	✓
InternetBeans Express employ proven DataExpress components for ultra-fast database access to JDBC™ or Borland® JDataStore™ databases	✓	✓
More than 15 InternetBeans for page producer, table, field, check box, combo box, text area, and more controls	✓	✓
Rapidly create dynamic, data-driven thin-client Web documents, including support for markup languages such as HTML, XML, and WML	✓	✓
Quickly link Web front ends with EJB		✓

## Premium tools

### Crystal Reports for Borland JBuilder

Custom report designer to access, format, and integrate data into JSP pages using an intuitive design environment		✓
100% native Java Reporting Components to process and render dynamic reports from within a J2EE-compliant application server		✓
Custom Tag Library to reduce the amount of coding required to embed reports into JSP pages		✓
Java Viewer API for providing customized report interaction—including viewing, drill-down, exporting and printing—from within J2EE applications		✓

### eBay SDK for Java

Create an application without the need to deal directly with SOAP, WSDL, XSD, XML, HTTP and SSL protocols		✓
API, Integration and Attributes Libraries to help streamline your development process		✓
Technical documentation (HTML and PDF) including a Getting Started Guide		✓
Java sample code including a robust eBay selling application		✓

## Borland® JBuilder® 2005 Feature Matrix

FOUNDATION      DEVELOPER      ENTERPRISE

Fortify Software for Borland JBuilder		
Source code security analyzer locates hard-to-find security flaws within Java/J2EE code automatically, including SQL injection, cross-site scripting, input validation vulnerabilities, plus many more		✓
Unique global data flow analyzer details exploit paths and pinpoints vulnerable functions accessible by tainted input		✓
Secure coding rules can be added and updated for more comprehensive coverage		✓
Integrated help and special on-line pages provide additional information on security vulnerability types, exploit patterns and remediation steps		✓

Sybase EAServer integration module		
Deploy and debug J2EE components in EAServer		✓
Create a test client to test EJBs running in EAServer		✓
Create CMP beans with EAServer-specific values gathered automatically		✓
Configure EAServer Profiles and post-deployment values via a new easy-to-use interface		✓

Vignette Portal plug-in for Borland JBuilder		
Help JBuilder developers quickly create, deploy and test custom portlets based on the Java standard (JSR 168)		✓
Create and configure portlets using intuitive wizards		✓
Rapidly create portlets ready to run with Vignette Application Portal Borland Developer Edition		✓
Provides full documentation and examples		✓

## Web Services development

Supported Platforms		
Apache Axis Web Services runtime (Apache Axis version 1.2)		✓
Support for WebLogic native Web Services runtime and deployment		✓
WebLogic 8.1 native Web Services		✓

Web Services designer		
Visual two-way Designer for creating Web Service-enabled applications		✓
Streamlined user interface and wizards		✓
WSI Monitor and analysis tool		✓
Support for Apache Axis 1.2 and BEA WebLogic 8.1 native Web Services		✓
Web Services flow design view		✓
Visually create and change Web Services components such as chains, handlers, request/response/fault flows and transports		✓
Visual configuration of complex types with serializers and deserializers		✓
Visual representation of WSDL and Java/EJB with methods and operations		✓
Activate or deactivate services without removing components from the designer		✓
Configure automatic client generation for services on project build		✓
Configure automatic server-side generation from WSDL on project builds		✓
Instant navigation between visual Web Services components in designer and source		✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Auto-scanning or on-demand scanning of session and message-driven beans to exporting as Web Services			✓
Drag and drop Java components or WSDL documents onto designer to create Web Services components			✓
Drag and drop from UDDI browser to designer to create Web Services components			✓
Dynamic validation and error insight to suggest missing configuration entries			✓
Auto-synching of Java components and Web Services components			✓
Form-based deployment descriptor editor			✓
Deployment descriptor code insight through XML code insight			✓
Support for asynchronous Web Services			✓
Drag and drop WSDL onto designer to generate Java classes			✓
Implement a Web Service as an EJB based on WSDL			✓
Drag and drop class or valid EJB component onto designer to expose selected methods of a Java class as Web Services and generate WSDL file			✓
Deploy an EJB as a Web Service			✓

### Build, debug, and run Web Services

TCP Monitor to monitor SOAP messages transported between client and server			✓
Create a SOAP or Axis deployment environment			✓
Dynamic generation and deployment of Web Services components during project builds			
Deploy Axis runtime to Tomcat, Borland® Enterprise Server, WebLogic, and WebSphere			✓

### UDDI Explorer

UDDI Web Services Explorer (which now supports UDDI 2.0) to browse and import available services			✓
Publish Java Web Service to UDDI directory			✓

## EJB development

### Supported platforms

Borland® Enterprise Server 6.0 Release Pack 1 with EJB 1.1/J2EE 1.2 and EJB 2.0/J2EE 1.3			✓
Borland Enterprise Server, AppServer Edition 5.2.1 with EJB 1.1/J2EE 1.2 and EJB 2.0/J2EE 1.3			✓
JBoss 3.0.8 with EJB1.1/J2EE 1.2 and EJB 2.0/J2EE 1.3			✓
JBoss 3.2.5 with EJB1.1/J2EE 1.2 and EJB 2.0/J2EE 1.3			✓
BEA WebLogic Server 8.1 SP3 with EJB1.1/J2EE 1.2 and EJB 2.0/J2EE 1.3			✓
BEA WebLogic Server 7.0 SP5, with EJB 1.1/J2EE 1.2 and EJB 2.0/J2EE 1.3			✓
IBM® WebSphere® 5.1.0.4 with EJB 1.1/J2EE 1.2 and EJB 2.0/J2EE 1.3			✓
IBM® WebSphere® 5.0.2.4 with EJB 1.1/J2EE 1.2 and EJB 2.0/J2EE 1.3			✓
IBM WebSphere 4.0.7 with EJB 1.1			✓
Sybase EAServer 5.1, 5.0, 4.2, and 4.1 <sup>1</sup>			✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
<b>Enterprise archives</b>			
J2EE application client packaging		✓	✓
Resource Adapters Archive (RAR) file support for packaging J2EE EIS connectors		✓	✓
Enterprise Archive (EAR) file support for packaging an entire set of components of an application; includes support for J2EE 1.2 and 1.3		✓	✓
EJB archives according to J2EE 1.2 specification			✓
EAR group file nodes J2EE 1.2- and J2EE 1.3-compatible			✓
Import existing external archives into EAR group			✓
Import from existing EJB groups			✓
EAR packaging, deployment, and run for all supported application servers			✓
J2EE application client JAR files			✓
<b>EJB test client wizard</b>			
Boundary testing			✓
Generate test clients from a prepackaged EJB archive			✓
Generate test clients based on JUnit or Cactus			✓
<b>DataExpress for EJB</b>			
DataExpress for EJB for seamless access to session beans and entity beans through DataExpress			✓
DataExpress access beans SessionBeanConnection, EjbClientDataSet, EntityBeanProvider, EntityBeanResolver to connect back-end business logic to front ends			✓
<b>Rapid J2EE platform-compliant development and deployment</b>			
Create multiple copies of application server integration configurations			✓
Incremental Deployment of classes to WebLogic 8.1			✓
Deployment to JBoss via exploded directories			✓
Customize synchronization of application server integration configurations with individual projects			✓
Extendable modular services-based server integration to support servlet, JSP, EJB, and deployment using multiple platforms.			✓
Local and remote Hot-Deploy deployment to running EJB container			✓
Java™ Messaging Service (JMS) wizard for creating message producers and consumers			✓
Generate or update source implementation skeletons from remote interfaces			✓
Visually create EJBs to comply with the J2EE standard			✓
EJB 1.1 and EJB 2.0 support with visual tools			✓
Visual creation of EJB deployment, environment, and control descriptors			✓
EJB modules to organize deployment descriptors and EJB archives			✓
Multiple EJBs per project			✓
EJB module wizard to create EJB groups for importing existing or creating new EJB descriptors			✓
EJB wizard for creating EJB 1.1 entity and session bean Java source code, including home and remote interfaces			✓
EJB interface wizard to simplify development of EJBs			✓
EJB test client wizard to create beans based on available EJB modules			✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Autopackaging of beans at project build time			✓
Creation of deployable archives per EJB group at project build time			✓
Instant execution of EJB in run or debug mode			✓
EJB 1.1-aware BeansExpress with visual property editor and method-level conflict resolution			✓
EJB test client wizard to create beans based on available EJB group			✓
EJB test client wizard to customize EJB client based on EJB group			✓
RAR support for packaging EIS connectors for J2EE			✓
Application client archive support			✓
JBoss Service Archive support			✓
EAR (application) support			✓

### Two-way deployment descriptor editor

Visual forms-based creation and configuration of XML deployment descriptors	✓	✓
Deployment descriptor XML source editor	✓	✓
Two-way editing of deployment descriptor for editing standard and vendor-specific descriptors for all supported servers		✓
CodeInsight for deployment descriptor elements via XML code insight	✓	✓
StructureInsight for visual and XML deployment descriptors	✓	✓
Support for Servlet 2.2, 2.3, and 2.4 and JSP 1.1, 1.1, 1.2, and 2.0 modules	✓	✓
Support for application client 1.2 and 1.3 modules		✓
Support for application 1.2 and 1.3 modules		✓
Support for connector 1.0 modules		✓
Support for EJB 1.1 and 2.0 modules		✓
Validator to validate standard descriptors for all supported module types	✓	✓

### Visual designer for EJB 2.0

Visual point-and-click two-way designer of entity, session, and message beans. Two-way generation of classes and deployment descriptors		✓
Improved performance, error handling, and better diagrams		✓
Support for more coding scenarios, including multiple ejbCreate methods per bean and bean inheritance, and two-way coding support for individual user coding styles		✓
Visually create Container Managed Persistence (CMP) entity beans, including properties and interfaces		✓
Visually design CMP 2.0 relationships and configure database mapping		✓
Visually create session beans		✓
Automatically arrange beans		✓
Create multiple design panes to logically group beans in an EJB group		✓
Generate database schema from entity beans in the designer		✓
Import EJB source code without deployment descriptors		✓
Import EJB descriptors, including vendor-specific descriptors, for all supported servers		✓
Control cardinality of CMP 2.0 relationships		✓
Automatically insert and update import statements when exposing a method with a remote interface		✓
Visually create message-driven beans		✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Edit bean package source paths			✓
Generate entity beans from existing database schema			✓
<b>JBuilder® Entity Bean Modeler™ for EJB 1.0 and EJB 1.1</b>			
User-defined mapping between table metadata and Java datatypes			✓
Generate BMP source code from table relations			✓
Entity Bean Modeler to create CMP and BMP entity beans, including home and remote interfaces, as well as primary key classes			✓
Combination of DataExpress integrated database tools with entity bean functionality			✓
Visual mapping of data source, tables, and fields to entity bean			✓
Model entity bean table relationships among tables for foreign keys, primary keys, unique indexes, field names, and field types			✓
Fine-tuning of class, interface, primary keys, and JNDI names for Java code generation			✓
Optionally generate EJB 1.1- or EJB 2.0-style classes with abstract field accessors			✓
<b>Borland Enterprise Server support</b>			
Borland Enterprise Server 6.0 Release Pack 1 Development Software (J2EE and CORBA) for JBuilder 2005 – a complete J2EE 1.3 and EJB 2.0 implementation, including both an EJB service and a servlet/JSP service			✓
Borland Enterprise Server, AppServer Edition 5.2.1 development support			✓
Support of EJB for stateless and stateful session beans, message-driven beans, and entity beans with bean-managed persistence and container-managed persistence		✓	✓
Dynamic Hot-Deploy, to deploy/undeploy/redeploy EJBs to container without shutting down or restarting			✓
JBuilder integration provides an authentic enterprise development and deployment experience for J2EE and Web Services applications			✓
Full support for the Java™ Connector Architecture			✓
Support for the CORBA 2.4 specification			✓
Support for distributed transactions with Java™ Transaction Service (JTS) and Java™ Transaction API (JTA)			✓
2PC Transaction Service			✓
Support for JMS through TIBCO Enterprise Message Service™ for JMS bundling			✓
TIBCO Enterprise for Java™ Messaging Service for JMS 3.1.1 integration			✓
Enhanced support for CMP 2.0 entity bean relationships			✓
J2EE Blueprints, including SmarTicket, PetStore 1.3, and clustered PetStore 1.3			✓
Support for local interfaces			✓
RMI-over-IIOP and Java-to-IDL mapping			✓
CORBA compliance with both RMI-over-IIOP and Java-to-IDL mapping			✓
Ability to instantiate CORBA objects and EJB in the same container			✓
Java Naming and Directory Interface (JNDI) naming, load balancing, fault tolerance, and failover in the Naming Service			✓
Both HTML-based and Java-based visual GUI tools for simplified administration			✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Security features out of the box, including strong 128-bit encryption, authentication, and authorization			✓
Support for efficient application partitioning			✓
Management functionality			✓
Integrated Management Console with auto-discovery of components			✓
Management server (Borland Management Hub) includes the ability to establish rules to create the desired state of managed applications			✓
Database Repository provides scalability and reliability			✓
Best-practices clustering support, including a Cluster Wizard to easily build a load-balanced cluster of J2EE servers			✓
Define rules to control start and stop sequence of services across multiple application servers in cluster			✓
Integrated Java Web server with support for servlets and JSP			✓
Web Services support based on Apache Axis—expose EJBs, Java, and CORBA as Web Services			✓
Apache 2.0 includes improvements in performance and scalability			✓
Tomcat 4.0 with metrics display to track number of hits, response time, and instance counts			✓
IIO Connector delivers improved performance, Smart Session support, and the ability to cluster Tomcats across multiple domains with failover and round-robin support			✓
Usability with the Borland Console allows easier configuration of Web container			✓

### BEA WebLogic Server™ support

Integrated BEA WebLogic Server™ 8.1SP3 (EJB 1.1 and 2.0, servlet 2.0 and 2.3, JSP 1.1 and 1.2) development support, including both an EJB service and a servlet/JSP service			✓
Integrated BEA WebLogic Server 7.0 (EJB 1.1 and EJB 2.0) development support, including both an EJB service and a servlet/JSP service			✓
Integrated support for WebLogic Server 7.0 and 8.1 native Web Services runtime			✓
Support for CMP and BMP entity bean creation			✓
EJB 2.0 CMP relationships for entity bean support			✓
EJB 2.0 message-driven beans			✓
Targeted deployment to WebLogic Server			✓
Dynamic Hot-Deploy to deploy/undeploy/redeploy EJBs to container without shutting down or restarting			✓
Integrated Two-Way-Tools for managing and deploying beans, including Deployment Descriptor Editor, Entity Bean Modeler, and more			✓
Configurable WebLogic ejbc/appc compiler settings for home and remote interfaces			✓
Support for exploded directory deployment			✓
Mapped EJB module support with support for redeploying individual classes from a mapped module			✓

### IBM® WebSphere® Server support

Integrated WebSphere 5.1.0.4 (EJB 1.1 and 2.0) development support, including both an EJB service and a servlet/JSP service			✓
Integrated WebSphere 5.0.2.4 (EJB 1.1 and 2.0) development support, including both an EJB service and a servlet/JSP service			✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Integrated WebSphere 4.0.7 (EJB 1.1) development support, including both an EJB service and a servlet/JSP service			✓
Support for BMP entity bean creation			✓
Targeted deployment to WebSphere server			✓
Integrated Two-Way-Tools for managing and deploying beans, including Deployment Descriptor Editor, Entity Bean Modeler, and more			✓

### Applications based on Session Façades, Struts, and DTO patterns

Session façade wizard to create session beans based on session façade and DTO pattern			✓
Wizard to create JavaBean client wizard for session façades			✓
Wizard to create Struts application for client-session façades			✓

## XML development

### Generic XML support

Project support for XML file types: XSD, XSP, DTD, XSL, MAP (XML-DBMS), WML, and XHTML	✓	✓	✓
XML TagInsight		✓	✓
Tag inspector		✓	✓
Browser-based XML viewer		✓	✓
Tree view for XML in browser view		✓	✓

### XML environment

Java API for XML Processing (JAXP) and DOM integrated library integration		✓	✓
Xerces 2.02 (XML) and Xalan 2.2 (XSLT) integration		✓	✓
Validation with error tracing		✓	✓
Transformation with source and browser view of XML and HTML output		✓	✓
Transform trace for selection, generation, and template creation		✓	✓
Cocoon publishing engine integration for creating XML Web applications			✓
XML-based Web application projects with Cocoon runtime integration			✓

### JavaBeans binding and persistence

Generate XML instance from DTD		✓	✓
Generate DTD from XML		✓	✓
Generate Java classes from DTD		✓	✓
Generate Java classes from schema		✓	✓
Open architecture for third-party, data-binding solutions		✓	✓

### Advanced XML support

SAX parser generator wizard, now based on JAXP			✓
XML ModelTableBean and XML ModelSQLBean to import/export DTD/XML data from or to relational databases			✓
XML TemplateTableBean and XML TemplateSQLBean to transform automatically generated XSL style sheets from XML to HTML			✓
XML mapping generation from RDBMS			✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
XML SQL generation from RDBMS			✓
XML SQLBean and XML TemplateSQLBean to generate XML packets complying with Borland XML DTD on declarative SQL templates			✓
Support for Castor 0.9.5.3, including source generation from schemas and custom binding files	✓	✓	✓

## GUI Development (AWT and Swing)

### Visual Java™ 2 two-way GUI designer

Visual Java™ 2 two-way designer for drag-and-drop JFC/Swing professional UI application development without proprietary markers or code	✓	✓	✓
Customizable toolbar	✓	✓	✓
Customizable code creation based on developer style preferences	✓	✓	✓
Visual customization of Java layout managers for rapid UI development	✓	✓	✓
Visual menu designer	✓	✓	✓
Visual design of JavaBeans components and integration of third-party components to the component palette	✓	✓	✓
IBM® VisualAge® for Java form designer support	✓	✓	✓
Support for layout managers, Swing components, and Swing borders	✓	✓	✓

### GUI coding productivity

Wizards for rapid development of applications, applets, JavaBeans, and more	✓	✓	✓
Application wizard to rapidly create GUI applications in Java	✓	✓	✓
Dialog, frame, and panel wizards with flexibility for Swing, AWT, or custom base classes to speed the creation of Java GUIs	✓	✓	✓
Applet wizard to quickly define and create Java applets	✓	✓	✓

## Database development

### Third-generation DataExpress components for simplified data access and management

Data module wizard to simplify data access development using data modules	✓	✓	✓
JBuilder® dbSwing™ data-aware JavaBeans for developing full-featured Java 2 database applications	✓	✓	✓
Swing data-aware models and binders for use with standard Swing components, including JTable, JList, and text controls	✓	✓	✓
Procedure DataSets to easily use stored procedure in applications	✓	✓	✓
Serializable DataSets for distributed database computing solutions	✓	✓	✓
DataExpress data-access components with JDBC database connectivity, including support for master-detail relationships, picklists, lookups, multiple table joins, and transactions	✓	✓	✓
JDBC data sources	✓	✓	✓
Connection pooling	✓	✓	✓
DataExpress source code for control and flexibility in building Java database applications			✓
Connect to registered data sources via JNDI	✓	✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
<b>Simplify database development with an integrated suite of SQL tools</b>			
Visual tools for creating two-tier client/server applications	✓	✓	✓
StructureInsight for SQL database queries		✓	✓
SQL Builder to visually create and edit SQL queries to JDBC data sources		✓	✓
Database Pilot™ for viewing database schema and creating connection URLs		✓	✓
Visual tools for database development, including column editor, connection editor, link editor, sort editor, MetaData Update Editor, and Resolve Order Editor	✓	✓	✓
SQL92 query syntax support	✓	✓	✓
Resourceable SQL to simplify management of SQL code	✓	✓	✓
JDBC monitor to test and tune SQL applications		✓	✓
Interactive SQL command-line tool		✓	✓
SQLJ syntax highlighting			✓
<b>JDataStore—embeddable Java™ database for development, prototyping, and deployment</b>			
Extremely fast JDBC database written entirely in Java	✓	✓	✓
Faster JDBC access than many established native databases	✓	✓	✓
Small RAM and disk memory footprint	✓	✓	✓
Generates very small database file footprint, often half the size of competitors	✓	✓	✓
Easier to embed and deploy using a single executable .jar file	✓	✓	✓
Nearly zero administration	✓	✓	✓
Automatic crash recovery	✓	✓	✓
Visual tools and error messages available in Japanese, German, French, and Spanish	✓	✓	✓
Development license for JDataStore included (additional licensing required for deployment)	✓	✓	✓
Extremely powerful! Supports complex business logic and thousands of users	✓	✓	✓
Tight integration between JBuilder and Borland Enterprise Server for highly productive development	✓	✓	✓
Certified on broad range of Java™ Virtual Machines (JVM™), including JVM 1.4, 1.3, 1.2, and 1.1	✓	✓	✓
Certified on many OS platforms, including Linux, Mac OS X, Windows® 98, Windows® 2000 Server, Windows NT® 4.0, Windows® XP, and Windows Server™ 2003, Enterprise Edition	✓	✓	✓
Supports small-device platforms, including Symbian OS,™ embedded Linux, and Windows CE®	✓	✓	✓
Passes the entire Sun J2EE 1.3.1 Certification Test Suite (CTS), including the JDBC, PM, XA, and JTA test trees	✓	✓	✓
Complete support for JDBC 3 compliance	✓	✓	✓
JDBC 3 save points and parameter metadata	✓	✓	✓
ODBC support for native applications using the EasySoft™ ODBC-JDBC Gateway	✓	✓	✓
Comprehensive subset of entry-level SQL92 standard	✓	✓	✓
Local Type 4 JDBC driver for high-speed in-process access	✓	✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Remote Type 4 JDBC driver for multiprocess/computer access	✓	✓	✓
Optimized for high transaction loads of J2EE/EJB/OLTP applications	✓	✓	✓
Optimized for complex DSS/OLAP queries	✓	✓	✓
Optimized to allow read-only transactions to operate at full speed without needing to acquire locks	✓	✓	✓
Ability to use read-only transactions for online backup and long-running reports for continuous database availability	✓	✓	✓
Support for all four JDBC transaction isolation levels	✓	✓	✓
Row-level locking for increased concurrency	✓	✓	✓
JDBC connection pooling with statement caching	✓	✓	✓
High-speed transactional crash recovery from system failures	✓	✓	✓
Unicode for String/Char data	✓	✓	✓
Support for collation keys for international sorting and indexing	✓	✓	✓
Database and log files portable across all platforms	✓	✓	✓
Support for many additional SQL keywords, including: ACTION, ANY, CALL, CHECK, EXCEPT, FOREIGN, INTERSECT, NO, REFERENCES, SOME, and USER	✓	✓	✓
Stored procedures and user-defined functions that can be implemented in Java	✓	✓	✓
Support for Java triggers	✓	✓	✓
Security features for user authentication and database encryption	✓	✓	✓
Rich data type support for Java primitives, Java objects, and BLOBs	✓	✓	✓
Data replication support using DataExpress components for disconnected computing models	✓	✓	✓
“Pluggable” storage for DataExpress data-access JavaBean components	✓	✓	✓
Direct binding of tables and columns to visual swing JavaBean controls	✓	✓	✓
Portable transactional file system capability for random-access files	✓	✓	✓
Visual JDataStore Explorer interactive data viewing and editing	✓	✓	✓
Visual JDataStore Explorer transaction, security, and backup administration	✓	✓	✓
Visual JDataStore Explorer import from other JDBC databases for interoperability	✓	✓	✓
Visual JDataStore Explorer interactive table/index create/restructure provided	✓	✓	✓
Visual JDataStore Explorer interactive SQL execution	✓	✓	✓
Visual JDataStore Server interactive database connection monitoring and logging	✓	✓	✓

## CORBA and RMI Development

### CORBA distributed object development

Included! Borland® VisiBroker® for CORBA® ORB™ development license, naming, and event services for creating distributed Java 2 applications	✓
Visual tools to rapidly create HTML front ends to CORBA servers using servlets or JSP	✓
CORBA interface wizard with sample interface definition to simplify the creation of IDL files	✓
CORBA server interface wizard to quickly define and create CORBA interface beans	✓
HTML CORBA client application wizard to quickly define and create a JSP or HTML client from an IDL file	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
CORBA server application wizard to rapidly create a CORBA server application from an IDL file			✓
Generation of server monitors to simplify testing of CORBA servers			✓
Advanced integration of CORBA/IOP development with support for both Borland VisiBroker for Java and Iona® OrbixWeb™ for Java data modeler to visually create IDL interfaces for distributed CORBA development			✓
IDL navigation and syntax highlighting			✓
Visual tools to rapidly create client applications for CORBA servers			✓
Visual tools to rapidly create CORBA servers			✓
Multi-tier CORBA sample applications using VisiBroker with source code			✓
Full support for the CORBA 2.4 specification			✓
Portable Object Adapter (POA) with support for POA policies, such as Thread, Lifespan, Object ID Uniqueness, ID Assignment, Servant Retention, Request Processing, and Implicit Activation			✓
Objects By Value (OBV) for passing arbitrary complex objects and graphs of objects by value across processes, machines, and languages			✓
Java2iiop compiler to create IOP-compliant stubs and skeletons directly from Java interfaces			✓

### RMI distributed object development

Integrated project management for RMI development	✓	✓	✓
Automatic invocation of RMIC postprocessor to create skeletons and stubs	✓	✓	✓
Synchronized interface and implementation			✓
RMI encapsulation			✓

## Mobile development

### Supported platforms

Supports MIDP 1.0 and MIDP 2.0 profile	✓	✓
Sun J2ME™ Wireless Toolkit 2.1 included	✓	✓
Support for Nokia® Developer's Suite for J2ME™	✓	✓
Support for Siemens® Mobility Toolkit (SMTK) for Java™ Development	✓	✓
Support for Sprint PCS Wireless Toolkit™	✓	✓
Support for Sony® Ericsson® J2ME™ SDK	✓	✓
Support for NTT DoCoMo™ DoJa™ 1.5 and 3.0	✓	✓

### Code productivity for mobile applications

MIDP project wizard	✓	✓
MIDlet wizard	✓	✓
Class/Package Browser	✓	✓
CodeInsight for CLDC/MIDP classes	✓	✓
Micro test case wizard	✓	✓
Micro test wizard to create a suite of test cases	✓	✓
MIDlet signing for use with digital certificates	✓	✓
Support for user-defined permission attributes	✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
<b>Two-way mobile visual designer</b>			
Two-way visual designer for rapid application development		✓	✓
Designer support for MIDP 1.0 and MIDP 2.0 UI elements		✓	✓
Screen manager to easily design complex, multiscreen applications		✓	✓
<b>Debug mobile applications</b>			
Debug in micro-device emulator		✓	✓
Debug on actual micro device (if enabled in device JDK)		✓	✓
<b>Build and run mobile applications</b>			
Mobile JDK switching		✓	✓
Archive builder to automatically create JAR, manifest, and JAD files		✓	✓
Option to automatically obfuscate source code as part of archive process. RetroGuard v1.1 included, or configure to use obfuscator of your choice		✓	✓
Implementation of Unified Emulator Interface (UMI) for easier integration with third-party emulators		✓	✓
Over The Air (OTA) Provisioning to upload MIDlet suites to FTP server without leaving JBuilder. Download and test in emulator, or run directly from remote server		✓	✓
<b>Unit testing infrastructure</b>			
<b>Unit test code productivity</b>			
Test-case wizard to create test-case class skeletons	✓	✓	✓
Test-case fixture support		✓	✓
Fixture wizard for JNDI			✓
Fixture wizard for JDBC			✓
Comparison fixture wizard			✓
Custom fixture wizard			✓
Test configuration support			✓
<b>Debug and run unit tests</b>			
Tight integration of JUnit	✓	✓	✓
Built-in test runner	✓	✓	✓
Unit test collector runner for automatic unit test collection, test suite generation, and running			✓
Test stack trace filter to exclude classes from traces			✓
Test case wizard to create logical groupings of unit tests			✓
Automatic test discovery to identify test cases or test suites			✓
Quick navigation to unit test failures in your source code	✓	✓	✓
Debugging of test classes	✓	✓	✓
Right-click on methods to debug individual methods without debugging entire unit test class	✓	✓	✓
Borland® Optimizeit™ Code Coverage integration with unit test runner (Optimizeit is included only in JBuilder Enterprise)	✓	✓	✓

## Borland® JBuilder® 2005 Feature Matrix

FOUNDATION      DEVELOPER      ENTERPRISE

### Enterprise unit testing

Cactus support (1.61 for J2EE 1.3)			✓
EJB test client based on Cactus			✓
EJB test client based on JUnit			✓

## Version management and team collaboration

### Supported version control systems

Borland® StarTeam® support		✓	✓
CVS support	✓	✓	✓
Rational® ClearCase® version 4.2 and 5.0 support		✓	✓
Microsoft® Visual SourceSafe® support (Windows platform only)		✓	✓

### Integrated team development

File decorations indicate version status of file as modified or checked in	✓	✓	✓
SmartDiff for differences (diffs) between files, independent of formatting changes		✓	✓
Revision browser to display history information of source versions	✓	✓	✓
Revision information on revision type, timestamp, author, and label	✓	✓	✓
Revert to previous source revisions	✓	✓	✓
Visual source-level display of diffs in distinct source revisions	✓	✓	✓
Diffs among different types of source revisions, including file buffers and local and backup files	✓	✓	✓
Revision browser to view source versions	✓	✓	✓
To-do list manager and to-do source comment tags	✓	✓	✓
Conflict resolution to reconcile source versions between workspace and repository now applies to all file types	✓	✓	✓
Conflict navigation in source editor	✓	✓	✓
Checkout from branch	✓	✓	✓
Status browser to show which files have pending changes	✓	✓	✓
Commit browser to show and commit files to VCS	✓	✓	✓
Set version label: per file, per branch, per project	✓	✓	✓
Synchronized scrolling between source and history pane	✓	✓	✓
Difference view of arbitrary files	✓	✓	✓
Refactoring-aware RenameNotifier to create a rename action to be displayed in the Commit Browser of version-management systems that support the renaming function	✓	✓	✓
Prompt to check out files needed to complete a refactoring	✓	✓	✓

### Version control management

Version control console	✓	✓	✓
Version control configuration to select server location, repository location, branch, and version selection	✓	✓	✓
Version control OpenTools API to tightly integrate third-party version control systems	✓	✓	✓
Commit browser framework now supports action changes on multiple files at the same time	✓	✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
<b>Library and SDK definitions</b>			
Configuration definition categories for the granular differentiation among user, team, and environment settings	✓	✓	✓
JDK definitions for project, user, or environment level	✓	✓	✓
Class, source, documentation, and required library settings per library definition	✓	✓	✓
Share library definitions of JDK locations among team members	✓	✓	✓
<b>Local file versioning</b>			
Local or network file versions	✓	✓	✓
Local file striping to store snapshot of project		✓	✓
Configurable history of source files on file system	✓	✓	✓
<b>CVS-specific</b>			
Create local CVS repository	✓	✓	✓
Lightweight administration and branching support, including the ability to create, switch, and merge branches	✓	✓	✓
Special update support based on revisions	✓	✓	✓
Modify version labels	✓	✓	✓
Check out selected revision	✓	✓	✓
CVS commit browser to resolve conflicts	✓	✓	✓
CVS support for status browser	✓	✓	✓
CVS support for commit browser	✓	✓	✓
CVS file locking support with edit, undo edit, and watches	✓	✓	✓
CVS server add-on to list projects and branches	✓	✓	✓
Create project from CVS repository	✓	✓	✓
Browse CVS modules in repository	✓	✓	✓
Browse repository using the Pserver protocol	✓	✓	✓
Reconcile local files with repository: read-only reconcile and read-write reconcile	✓	✓	✓
CVS local and remote file system, remote CVS server, and remote shell connections, now with port selection	✓	✓	✓
Direct protocol support for CVS	✓	✓	✓
<b>Borland StarTeam-specific</b>			
Configure StarTeam	✓	✓	✓
Launch Cross-Platform Client	✓	✓	✓
Set personal options for workspace, files, change requests, requirements, tasks, and topics	✓	✓	✓
Update the active file with changes from the repository version and merge the differences between them	✓	✓	✓
Add, remove, check in and check out files from the StarTeam repository	✓	✓	✓
Run the StarTeam visual diff tool	✓	✓	✓
Lock/unlock and revert to most recent checked out version	✓	✓	✓
Update project to synchronize all files in workspace with repository versions and merge differences	✓	✓	✓
Status browser	✓	✓	✓
Commit browser	✓	✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Place project into StarTeam server	✓	✓	✓
Pull project from StarTeam server	✓	✓	✓
Show pending refactoring options	✓	✓	✓

### Rational® ClearCase®-specific

View editor to select, create, and modify ClearCase views		✓	✓
ClearCase support for status browser		✓	✓
ClearCase support for commit browser		✓	✓
Checking file in and out of Version Object Base (VOB) view and undo checkout		✓	✓
Direct protocol support for ClearCase		✓	✓
Create project from VOB		✓	✓
Place project into VOB		✓	✓
ClearCase snapshots support		✓	✓
Unified Change Management (UCM) support		✓	✓
Launch ClearCase Explorer		✓	✓
Apply label		✓	✓
ClearCase user options		✓	✓
ClearCase merge manager		✓	✓
ClearCase find checkouts		✓	✓
Create a ClearCase VOB (Versioned Object Base)		✓	✓
Create and update view		✓	✓
Project explorer		✓	✓
Join UCM project		✓	✓
Native ClearCase tools support for file operations, including check in, check out, undo checkout, add file, properties for file, history for file, compare with prior version, and version tree for file		✓	✓

### Microsoft® Visual SourceSafe® (VSS)-specific features

VSS commit browser to resolve conflicts		✓	✓
Directory browser to pull and place a VSS project		✓	✓
Pull and post files from and to VSS project		✓	✓
VSS support for status browser		✓	✓
VSS support for commit browser		✓	✓
Direct protocol support for VSS		✓	✓
Create project from VSS		✓	✓
Place JBuilder project into VSS		✓	✓

## Requirements management

### Supported requirements management systems

Borland® CaliberRM™			✓
---------------------	--	--	---

### CaliberRM™-specific features

View requirements in the JBuilder message pane			✓
Insert CaliberRM requirements comments into JBuilder code			✓
Configure filters to view a subset of requirements			✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Real time embedded requirements management interface			✓
Automatic commenting of code, driven by requirements			✓

## Complete performance tuning with Borland® Optimizeit™ Enterprise Suite 6

### Flexibility to match many development environments

Borland® Optimizeit™ Suite delivers complete performance management capabilities for identifying and solving code-level performance issues; this integrated solution comprises Borland® Optimizeit™ Profiler, Borland® Optimizeit™ Thread Debugger, and Code Coverage			✓
Improve performance and reliability of any Java code: Java applications, J2EE applications, servlets, applets, EJBs, JavaBeans, JSP applications, and Java tag libraries			✓
Requires no code modifications. Runs with Sun JDK 1.1.8, most Java 2-compliant JVMs, including J2SE v1.3 with built-in HotSpot Client Virtual Machine, J2SE v1.4.1, IBM® Developer Kit, and Java™ 2, Technology Edition, version 1.3			✓
Integrates fully with popular application servers, including Borland Enterprise Server, IBM WebSphere, BEA WebLogic, and Sun ONE			✓
Seamless integration with JBuilder enables developers to run Optimizeit Suite right from within the JBuilder UI	✓	✓	✓
Easily connects to a remote Java process to test a program running on a different machine			✓
Optimizeit Progress Tracker analyzes the performance impact of code changes by comparing two or more snapshots			✓
Graph view in Optimizeit Progress Tracker plots performance variations over time			✓

### Optimizeit Profiler for memory and CPU profiling

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		✓	✓
Object Size Display automatically computes and displays, in real time, the amount of memory being consumed by all instances of a class; view and sort by object size to prioritize objects consuming the most memory		✓	✓
Reduced reference graph provides a transitive closure of the full reference graph to display only references that should be removed in order to free the object for garbage collection		✓	✓
CPU Profiler: Measure pure CPU usage or time usage during a profiling session, with option to use sampling-based or instrumentation-based profiler		✓	✓
Displays profiling information per thread and thread groups, with color highlighting of threads that were busy during profiling session		✓	✓
HotSpot Display lists methods where most time was spent, to help identify bottlenecks due to single methods		✓	✓

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
New scalable call graph visually isolates critical code. . Select a string allocation and highlight the flow of a method call to see where memory and time are being spent		✓	✓
Automatic Application Quality Analyzer supports performance-error prevention and coding standards by automatically detecting VM-level performance bugs		✓	✓
PDF Reporting: Generate reports in PDF to capture performance data		✓	✓

### Optimizeit Thread Debugger

Real-time display of the progress of all threads running within the Virtual Machine			✓
Easy-to-read color coding of thread behavior makes it easier to understand thread issues occurring in real time: thread contentions for a monitor, thread starvation for a resource, and excessive locking			✓
Understand thread contentions for a monitor with the detailed panels of the Contention View			✓
Wait state monitoring: understand why a thread is not making progress with the Waiting View and I/O Waiting View			✓
Track excessive locking: Monitor Enter View reports where a thread enters and holds monitors, allowing developers to understand and correct excessive locking			✓
Analyze Deadlocks: graphical view of the relationships between threads and monitors to quickly understand deadlock situations			✓
Predict deadlocks: Monitor Usage Analyzer generates full list of warnings and errors that might lead to deadlocks and performance bottlenecks, such as lock order warnings, lock and wait warnings, and lock and I/O wait warnings			✓

### Optimizeit Code Coverage

Real-time Class Coverage View to quickly see the coverage for each class and identify classes not fully covered			✓
Real-time display of all classes and interfaces used by the tested program and real-time percentage of lines covered per class			✓
Method Coverage View displays the methods and lines of code for a selected class that have not been used, allowing developers to modify test plans to cover all areas of the code			✓
Source Code Viewer shows lines of code that have never been executed, making it easier to spot dead code			✓
Option to display the interfaces that have been loaded by the virtual machine and those that have not been loaded			✓
Batch-mode support to easily include code coverage in any batch-mode testing process			✓
SnapshotMerger merges different runs of the same application from the command line to see coverage of the combined runs			✓
Generate customized, detailed coverage reports directly from the command line			✓

### Optimizeit Request Analyzer

Profile the performance behavior of your J2EE application code across the following J2EE components: JDBC, JSP, JNDI, Enterprise JavaBeans, and JMS containers			✓
--	--	--	---

## Borland® JBuilder® 2005 Feature Matrix

	FOUNDATION	DEVELOPER	ENTERPRISE
Improve performance and reliability of J2EE-related application code earlier in development with drill-down performance information for J2EE components			✓
Visual interface simplifies the complexity of J2EE application interactions using graphical representation			✓
System Dashboard view provides a graphical display of the application time spent in J2EE components and total number of requests. Shows the percentage of use for each server module to quickly detect any major component-level performance issues			✓
System Composite view displays all of the J2EE events that have occurred in an application, in real time, in their proper hierarchy. Hierarchy shows the relationship of events in terms of which events spawn others			✓

Made in Borland® Copyright © 2004 Borland Software Corporation. All rights reserved. All Borland brand and product names are trademarks or registered trademarks of Borland Software Corporation in the United States and other countries. CORBA and ORB are trademarks or registered trademarks of Object Management Group, Inc. in the U.S. and other countries. Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries. Microsoft, Windows, and other Microsoft product names are trademarks or registered trademarks of Microsoft Corporation in the U.S. and other countries. All other product names are trademarks of their respective companies. Offices in Australia, Brazil, Canada, Czech Republic, Finland, France, Germany, Hong Kong, Hungary, Ireland, Japan, Korea, Mexico, the Netherlands, New Zealand, Singapore, Sweden, Taiwan, the United Kingdom, and the United States. • 22587