Model-Driven Development with the UML

Rational Rose for Java Developers

You probably have experienced first hand the impact of today's fast-paced software development. You face accelerated development cycles and you must deliver high-quality software. Trade-offs are a thing of the past—you cannot sacrifice software quality for an on-time release, nor can you drop features to meet deadlines. If you fail to meet both challenges, you put your organization at risk of losing a competitive edge.

This leaves your development team working faster and harder than ever, typically coding every feature and detail of your application through a time-consuming process. The solution? Work at a higher level to specify the software architecture, while minimizing repetitive, manual development tasks. This is precisely where Rational Rose® comes in to accelerate development, improve quality and ensure architectural flexibility. Tailored for easy and efficient developer use, Rational Rose helps you build better, more robust software faster than ever before.

Model-Driven Development with the UML

The award-winning Rational Rose visual modeling tool is renowned for its ability to help build adaptable, robust software architectures that are easily changed to meet new business needs. Rational Rose streamlines development by automatically completing development tasks such as code generation and test asset creation. It scales to support teams of all sizes, and is part of Rational® Software's integrated software lifecycle development solution. Rational Rose now includes even stronger support for Java™-based development.

Unique model-driven development capabilities get new software development projects underway more quickly. You work above the details, performing more development tasks at the model-level and generating more parts of the application from these models. By focusing on the model and automatically generating parts of the application, you minimize hand-coding, automatically implement

language and platform-specific elements, and can even streamline Enterprise JavaBeans™ component testing. This makes it easier to work with Java technologies and reduces the number of manual steps. You can adapt designs for target deployment architectures and base solutions on industry standards such as J2EE™, J2SE™, XML, and CORBA.

The ability for deploy-to-test within Rational Rose in Windows provides the functionality to deploy Java and J2EE applications directly from Rose to your J2EE application server. The servers include IBM WebSphere, BEA Web Logic Server and the Sun Java Reference Implementation. It also allows developers to remain in Rose while deploying, and then leverage Rational QualityArchitect to test EJBs and/or run the web components in a unit test mode.

Unifying Development Teams

Rational Rose unifies all members of your development team through modeling of a variety of application areas, including software, databases, and Web pages in one tool and one language. Visual modeling in Rational Rose is done in the Unified Modeling Language™ (UML), the standard notation for software development across the lifecycle.

When analysts, application and data modelers, and other development professionals use Rational Rose to define aspects of the application, communication flows more naturally and the potential for error is reduced. Requirements, the system architecture, code and components, and databases can all be defined and inter-related in the UML, resulting in significant reduction of manual mapping activities. For example, software developers and system architects have ready access to all models and can use this valuable information as they construct solutions. By having the entire team communicating in one language and one tool, you minimize costly-to-resolve errors that are caused by inadequate communication.

Rational®
the software development company

Model-Driven Development with the UML

Creating a Robust System Architecture

Architecting applications in Rational Rose helps reduce the time-consuming and costly re-writes associated with code-only development. Through visual modeling, you ensure a robust application platform that responds easily to change. Because you visualize the application without focusing on details, the system and requirements are refined before committing to code. The result: well-architected, adaptable applications that scale to meet business and technical demands.

Rational Rose supports the complete set of diagrams specified in the UML, including use cases, sequence, collaboration, class, component and deployment diagrams, data models and others. Use as many types of diagrams as needed — it's easy to add additional diagrams later with a full UML modeling solution like Rational Rose. When you need a practical process to follow, Rational also supports you with best practices for software development and UML modeling.

One Tool for All Technology Needs

Only Rational Rose is flexible enough to support a wide range of development technologies. Integration with legacy systems, the need for early and frequent testing, and support for multiple IDEs are just a few of the challenges you face. Rational Rose provides powerful solutions to these problems, allowing you to reverse engineer code into visual models and generate code from models in multiple languages. Through an innovative feature called Rational® QualityArchitect, you can leverage models to begin early testing of EJB components —

mitigating risk, improving predictability, and avoiding downstream problems.

Rational Rose works with all of your favorite IDEs, so you can be sure that one tool meets an array of technology needs. Rational Rose supports J2*E technolgies and you can also generate, ASPs, CORBA IDL, XML DTDs, C/C++ code, and SQL and DDL for ODBC-complaint databases. Working in one flexible tool, there is no need for re-training or re-tooling as the environment expands and technologies change.
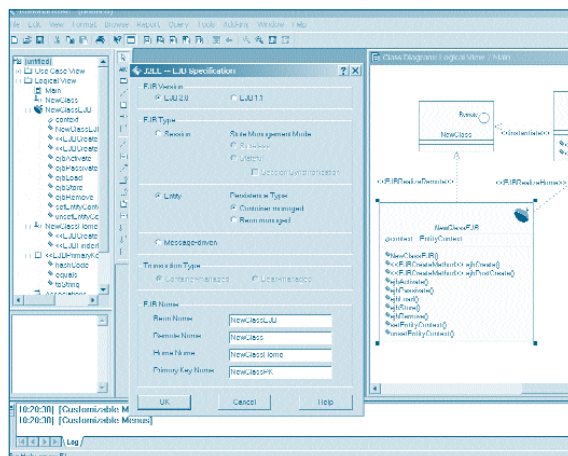
And with over 100 different Rose-link and add-in products available, Rational Rose is even more flexible including the ability to generate platform-specific Java code for popular Web server environments. Other products in the Rational software development solution are tightly integrated with Rational Rose to provide related functions such as requirements management, best practices, and many others. Rational Rose is the single solution you need to meet the demands of development projects today and over the long term.

A Complete Environment for Java Development

Working with Java, EJBs and Servlets is a simplified experience in Rational Rose's powerful round-trip engineering environment. You can model systems that target any or all of the Java 2-compliant technologies, and automatically generate buildable Java code from within Rational Rose's round-trip engineering environment. Rational Rose supports the UML Profile standard for specifying EJBs.

Rational Rose helps you align system design with business and user needs. You can base visual models on project requirements, easily updating use cases and other models as requirements change. Through tight integration with Rational's requirements management solution, you are assured that you are always working with the latest set of requirements, even when they change.

Once visual models are complete, it's time to generate more complete and predictable code. Servlets, EJBs, JSPs, XML and other components can all be generated through a few mouse clicks in Rational Rose. Rational Rose can also apply design patterns — including support for EJBs and "Gang of Four" Patterns — to generate code that implements behavior. The development cycles saved through code generation can be applied to new software features or more challenging tasks.

Rational Rose generates EJB code in version 1.1 or 2.0 through its streamlined EJB code generation interface.

To ensure a smooth transition between modeling and coding environments, Rational Rose is tightly integrated with leading Java IDEs including IBM® VisualAge® for Java, Sun Forte® for Java, Borland's JBuilder® and WebGain VisualCafe®. When you change the code or model, you can see the other being automatically updated. This tight integration combines the power of the leading visual modeling solution with leading Java IDEs to speed development processes.

Round-trip engineering offers a range of options for synchronizing models and code. For applications where models don't exist, Rational Rose lets you reverse engineer Java code, EJBs, Servlets, JSPs and XML files into visual models so that you can visualize the application and verify the architecture. You can then change the code, the model or both, and easily re-synchronize them.

Only Rational Rose's controlled synchronization options put you in complete control of when and how to synchronize your Java code and models. You can mix and match manual and automatic- synchronization techniques at your option. Use the autosynchronization feature to automatically synchronize an entire model with the corresponding code, or some elements of the model with the code. Once the elements to synchronize are established, this feature operates automatically. Autosynchronization works with code from any of the supported IDEs, synchronizing models-to-code or code-to-models. Or if you prefer, you can use manual synchronization. Designate all or parts of a model or code files and perform a one-time synchronization at your command.

An embedded code editor is included in Rational Rose, allowing you to view both the code and the model at the same time.
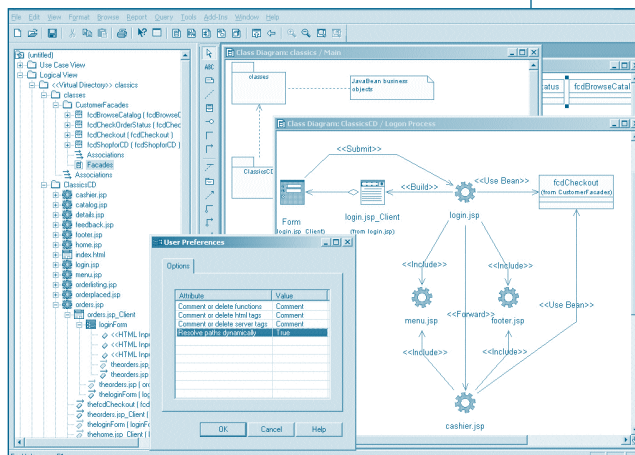
Java Development Support

Rational Rose includes the following features that accelerate Java development while ensuring software quality:

- **EJB 2.0 Support** – In addition to EJB 1.1 support, Rational Rose now generates EJBs that are fully compliant with the new EJB 2.0 specification, including Message-Driven Beans and the new container-managed persistence mechanism for Entity Beans.

- **Gang of Four Patterns** – Rational Rose allows you to optionally apply these well-known patterns to Java classes when Java code is being generated. These pre-defined patterns cover areas such as model-to-model

expansions, expanding classes, generating code, adding additional Java-specific methods, and code injections.

- **Deploy and Test** – The deploy-to-test add-in within Rose provides the functionality to deploy Java and J2EE applications directly from Rose to your J2EE application server.

- **Javadoc Support** — Rational Rose supports the creation of Javadoc tags. You establish a global policy defining which of the applicable tags should be generated for each type of code construct, such as class, or method. You can incorporate both standard and user-defined tags.

- **Simplified EJB and Servlet GUIs** — Rational Rose's interfaces for the generation of EJB and Servlet code is a streamlined mechanism that includes helper functions for adding items like business methods, and makes it easier to work with Java code.

Quality By Design: Build Quality into Your EJBs Throughout The Project Lifecycle

Visual modeling of JSP pages in Rational Rose.

Testing EJB components presents special challenges when related components are not yet available. You want to test components early when you can more easily resolve problems. Until now, this left you using precious development cycles to manually write test harnesses. Available in Rational Rose Enterprise, Rational QualityArchitect is a Rational Rose feature that automates this time-consuming process by generating drivers and stubs directly from visual models. These automatically-generated drivers and stubs work with EJB and Java components so that you can run tests to gauge how completely requirements are met. From test results, you can fine-tune functionality even before the

application has been built. Rational QualityArchitect is another way that Rational Rose helps you deliver with speed and quality while minimizing manual tasks.

Proven for Project Success

Optimal project success is ensured when you combine Rational Rose with other Rational or third-party solutions that support the development process. Rational Rose is a key element of Rational's software development solution and is tightly integrated with Rational's tools as well as third-party development tools to speed development, improve communication, and ensure software quality and accuracy. Rational Rose works with third-party tools such as leading IDEs, Microsoft® Visual SourceSafe™, and SCC-compliant version control systems.

Rational Suite: A Complete Lifecycle Solution

Rational Suite® is Rational's complete, integrated lifecycle development solution that unifies the cross-functional development team while it optimizes productivity. There is a solution that is right for system analysis, software development, system testing, content management, and project management.

Rational Rose is included in Rational Suite Enterprise, Rational Suite® DevelopmentStudio and Rational Suite AnalystStudio™. Each edition combines a full set of tools for that function and the Rational Suite Team Unifying Platform, a proven foundation for unifying cross-functional teams.

Put products in the Rational software development solution and Rational Suite Team Unifying Platform to work for you:

Rational Unified Process™ improves team process with easy, browser-based access to industry-proven software development best practices.

Rational RequisitePro® offers team-based requirements management that allows your team to fully understand the impact of change.

Rational® TestManager provides a single focal point for managing test activities and assessing software quality.

Rational ClearQuest® provides comprehensive

change and defect tracking that easily adapts to the way you work.

Rational ClearCase® LT manages your evolving software with the world's leading software configuration management solution.

The Rational Developer Network keeps you informed of the latest industry topics with a wealth of white papers, articles, book reviews, FAQs, industry news, and more.

When you combine Rational tools with our world-class consulting, training and support services, you have the most complete software development solution available.

The Choice for Software Development

The award-winning Rational Rose leaves your team unified and working smarter. Strongly supporting Java development, Rational Rose accelerates processes, ensures architectural quality, and streamlines manual tasks. Rational Rose positions you for today and tomorrow, whether you are working in Java or other technologies. Find out how Rational Rose can help you build better software faster. Visit www.rational.com/products/rose to download evaluation software and demos, request a free CD on the UML and much more.

Accelerate Success with Rational Software Development Services

Like all Rational tools, Rational Rose is supported by an extensive, worldwide service organization. Rational Services can help you build your team's capability through training, consulting services, and technical support. Our services organization offers over 60 courses and a variety of implementation and support plans to speed technology deployment and accelerate project delivery. Available where and when you need them, Rational services improve self-sufficiency as they build a foundation for continuous software development improvement.