

IBM Rational Software Modeler, Version 7.0

Highlights

- **Offers easy adoption and use**
- **Provides a simplified user interface**
- **Built on the open, extensible Eclipse 3.2 platform**
- **Offers extensive support for UML 2 modeling**
- **Includes model-to-model and model-to-code transformation development tools**
- **Supports the development of custom metamodels**
- **Automates traceability from requirements to design**

Modeling reduces the risk associated with developing systems by allowing organizations to specify and communicate information about development projects from several perspectives and to a variety of stakeholders. Modeling tools can also automate repeatable activities and help improve the productivity and overall maturity of the development process. That's why system architects, system analysts and system designers leverage the capabilities of the IBM Rational® Software Modeler product to help visually model and design their systems.

Rational Software Modeler, Version 7.0 software includes extensive support for the Unified Modeling Language, Version 2.1.1 (UML 2), the industry's standard notation for modeling a diverse range of software systems. The Rational Software Modeler product is built on top of the open and extensible Eclipse 3.2 platform. Together, these tools provide an unprecedented level of development environment flexibility.

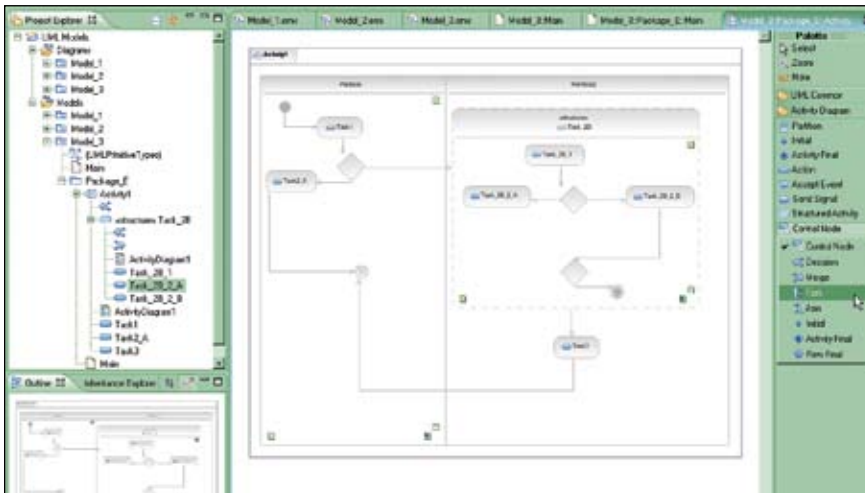


Figure 1: Rational Software Modeler software provides rich support for UML modeling based on UML, Version 2.1.1.

This Rational product includes the ability to apply patterns and transformations, enabling development teams to leverage team members' work, apply best practices, develop a sound system architecture and increase productivity. Extended model search, analysis and validation capabilities, as well as powerful model exploration tools such as *Browse diagrams*, round out the productivity features.

Leverage an easy-to-use tool

It's hard to incorporate new technology into existing processes, so IBM has designed flexibility options into products like Rational Software Modeler to help make them easier to use—beginning with installation. A flexible installation process provides a choice of options at a granular level, so that each member of your development team has access to only the necessary functions. Migration wizards are included to help you migrate to Rational Software Modeler from prior-generation IBM modeling products such as IBM Rational Rose® and IBM Rational XDE™ software.

Model applications more productively than ever with UML 2

The Rational Software Modeler solution, based on the UML 2, includes new ease-of-adoption and ease-of-use features that raise the bar for user productivity in modeling workflows. Its support for a variety of diagram types aids in design, discovery and documentation activities, including freeform diagram support. The UML 2 modeling capabilities in the product include new object diagram support and extended capabilities for modeling activities, composite structures, components,

interactions (sequence and communication diagrams) and state machines—in addition to providing use-case, class and deployment diagramming functions.

Exploit the latest in modeling-language standards

Through experience, modeling practitioners have found that the 1.x versions of UML lacked sufficient expressiveness in certain areas, such as modeling complex systems, where data, application and presentation logic needed to be separated to reduce complexity. And even those most experienced with UML needed better guidance for separating business and application logic from underlying implementation technology.

The Object Management Group's (OMG's) answers to these needs are found in UML 2 and the Model Driven Architecture (MDA) initiative, both of which allow users to express their architecture with more clarity and

control than ever before. MDA provides process guidance and higher degrees of expressiveness in modeling complex systems. The Rational Software Modeler product supports the MDA initiative by allowing the user to define multiple levels of models coupled with user-defined transformations between models and code, resulting in a clearer separation of concerns across the lifecycle.

Modeling for teams

With the Rational Software Modeler product, model files can be deconstructed into smaller components and placed under version control at a more modular level to better support parallel development activities. Its sophisticated model-merging capabilities let you compare, merge and reconcile models after parallel development is complete. Even more sophisticated model-fusing capabilities let you leverage existing model components to create new logical model structures.

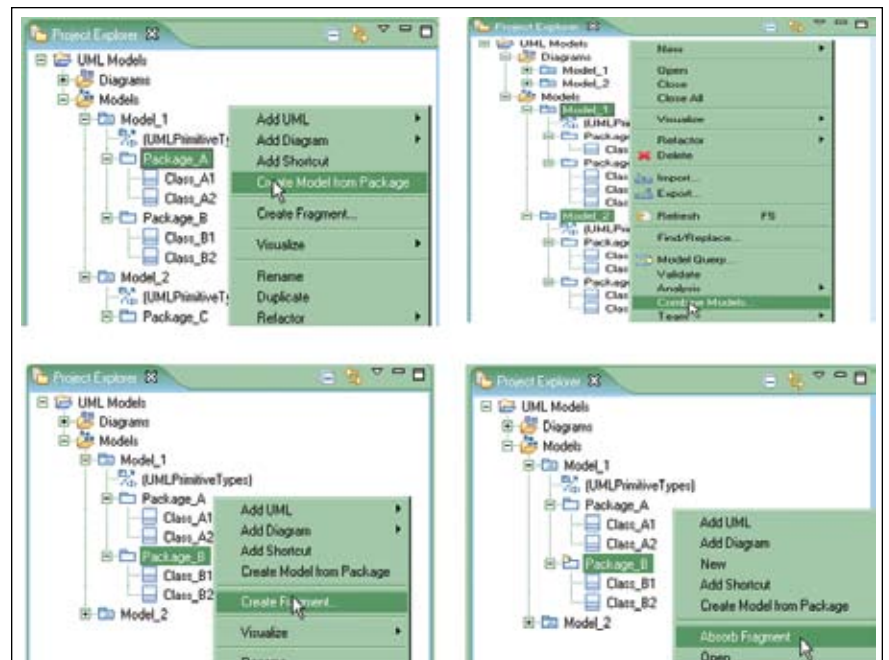


Figure 2: Rational Software Modeler software affords highly flexible model management. Logical models can be broken apart and combined, as well as partitioned into fragments (multiple physical files) that can be independently version controlled.

Tap into the power of model-based automations

Modeling alone adds value and helps reduce project risk, but even greater benefits are realized when models are used to automate the creation of other development artifacts, including other models, code and more. Use the included design patterns to help you more quickly build the content of your UML analysis and design models. Or, create your own UML-based patterns to extend this benefit even further.

The models that analysts and architects create also work with the IBM Rational Systems Developer and IBM Rational Software Architect model-driven development products. These products include UML-based transformations for generating Java™, Enterprise JavaBeans (EJB), C++, CORBA and Web Services Description Language (WSDL)/XML languages. A forthcoming product will extend the Rational Software Modeler product, with transformations from UML to C#, and from C# to UML.

The Rational Software Modeler product itself includes a transformation for generating Logical Data Models (LDMs) that are supported by the IBM Rational Data Architect product, thus enabling database designers to more easily leverage the software architecture into the data designs. And the Rational Software Modeler product includes tools for developing custom transformations that might target any type of implementation outputs and transformations between UML models at different levels of abstraction.

And because patterns and transformations can be used to automate repetitive development tasks and promote reuse and best practices, they can help to achieve more repeatability and lower defect rates.

Take advantage of the Eclipse open and extensible platform

For many years, software developers have realized and used the benefits of open standards in their projects. Now, through Eclipse, all of the benefits of open software have come to the development environment itself. Because the Rational Software Modeler product is built on top of the open Eclipse 3.2 platform, you can more easily extend its features to meet specific project requirements through an ecosystem of third-party plug-ins. Eclipse is written in Java code, which means that you can outfit your team across both Microsoft® Windows® and Linux® desktops.

Because the product features of the Rational Software Modeler solution and the Eclipse foundation are modular, installation is highly configurable. Install both the Eclipse environment and Rational Software Modeler product features. Or install just selected Rational Software Modeler features into an existing Eclipse shell that, for example, supports an established development project or environment. This modularity also makes it possible to unify the data, application and software development environments on Eclipse.

Integrate with other facets of the lifecycle

Complex software projects need traceability throughout the lifecycle. When requirements change, architects need to know what part of the architecture is affected. Such projects also require management of change, as it impacts model files and other lifecycle artifacts. This all becomes quite complex and challenges even the best-managed change processes. Failing to address these issues introduces increased risk to overall project success.

The Rational Software Modeler product helps you integrate with other facets of the lifecycle. Requirements stored and managed in IBM Rational RequisitePro® software can be accessed, associated to corresponding modeling elements and synchronized with user-selectable rules. Users can generate reports highlighting traceability from requirements to design. IBM Rational ClearCase® LT software, IBM's robust software configuration management product, can be used to manage model files. Alternatively, the Rational Software Modeler product integrates with Concurrent Versions System (CVS) for users who are already committed to that tool.

Integration with IBM Rational SoDA® software means that project documentation can be generated from the visual models. And the integration with the IBM Rational Unified Process® (IBM RUP®) solution gives teams the ability to work through all of this with common, online and integrated process guidance. These integrations reduce the risk associated with software development and make application development more predictable.



© Copyright IBM Corporation 2006

IBM Corporation
Software Group
Route 100
Somers, NY 10589

Produced in the United States of America
12-06
All Rights Reserved.

ClearCase, IBM, the IBM logo, Rational, Rational Rose, RequisitePro, Rational Unified Process, RUP, SoDA and XDE are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft and Windows are trademarks of Microsoft Corporation in the United States, other countries, or both.

Other company, product and service names may be trademarks or registered trademarks or service marks of others.

The information contained in this documentation is provided for informational purposes only. While efforts were made to verify the completeness and accuracy of the information contained in this documentation, it is provided "as is" without warranty of any kind, express or implied. In addition, this information is based on IBM's current product plans and strategy, which are subject to change by IBM without notice. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this documentation or any other documentation. Nothing contained in this documentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM (or its suppliers or licensors), or altering the terms and conditions of the applicable license agreement governing the use of IBM software.

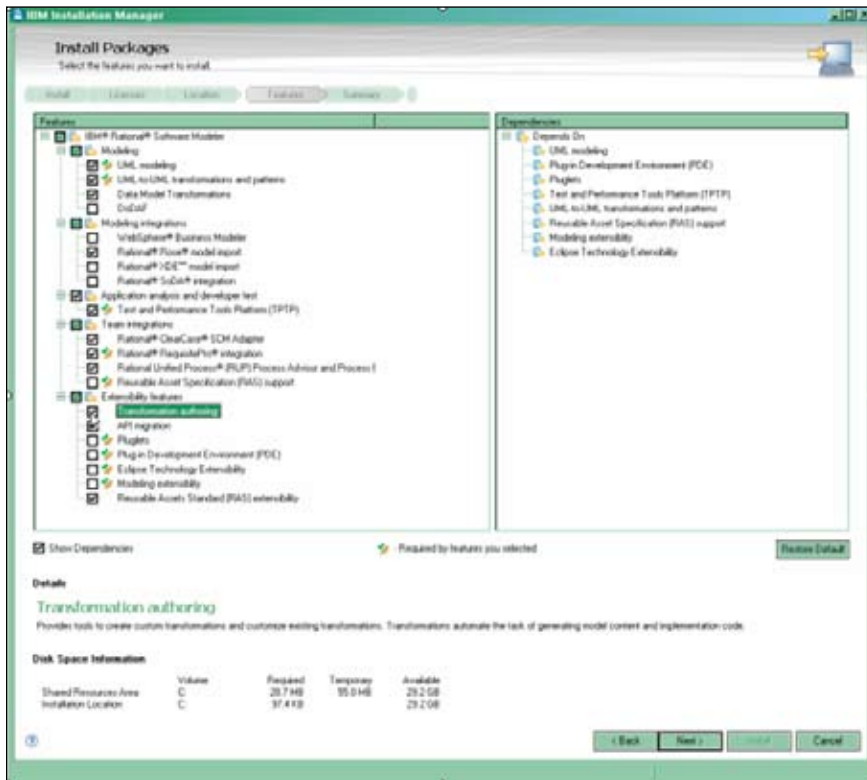


Figure 3: IBM Installation Manager enables installation of Rational tools (including Rational Software Modeler software) into existing Eclipse instances, and it supports the optional installation of product features.

Leverage a component of the IBM Rational Professional Bundle

The Rational Software Modeler product is a component of the IBM Rational Professional Bundle—a solution that brings together IBM Rational design, development and test products. This bundle includes all of the desktop tools your enterprise needs to design, construct and test Java Platform, Enterprise Edition (Java EE), portal and service-oriented applications on Windows and Linux platforms, and to test Microsoft .NET applications. The bundle provides a single-purchase vehicle with just one maintenance contract to manage—and it's all based on the flexible, open Eclipse environment.

For more information

To learn more about IBM Rational Software Modeler software, or to download a trial version, visit:

ibm.com/rational/adc