IBM Rational Systems Developer, Version 7.0

Highlights

- ***Offers integrated design and development, accommodating visualization and editing of Java and C++ structure and behavior via UML diagrams***
- ***Enables architectural modeling and specification with support for various programs, patterns and products***
- ***Offers easy adoption and use***
- ***Provides an open and extensible modeling platform***
- ***Is designed to integrate with your software lifecycle and development team***

Developing systems in C/C++ and Java™ Platform, Standard Edition 2 (Java SE 2) is becoming increasingly complex. Yet, developers are expected to deliver capabilities faster and more reliably than ever before. Modeling and other visualization technologies help improve developer productivity and software quality, especially when such features are integrated directly into a development environment consisting of flexible and powerful tools.

IBM Rational® Systems Developer, Version 7.0 software is an integrated design and development product that helps systems architects model complex systems, and helps software architects and systems developers create software-level system components using Java SE, C/C++ and CORBA languages. Rational Systems Developer software is built on the Eclipse platform, so it brings all of the benefits of open technology—including tool integration and increased flexibility—to the development environment.

Leverage an easy-to-use tool

It's hard to incorporate new technology into existing processes, so IBM has designed flexibility options into its Rational software to help make products like Rational Systems Developer software easier to use—beginning with installation. Its flexible installation process provides a choice of options at a granular level, enabling each member of your development team to install only the necessary features and functions in order to increase productivity. Migration wizards are included as well to help you move to Rational Systems Developer software from prior-generation IBM modeling products such as IBM Rational Rose® and IBM Rational XDE™ software.

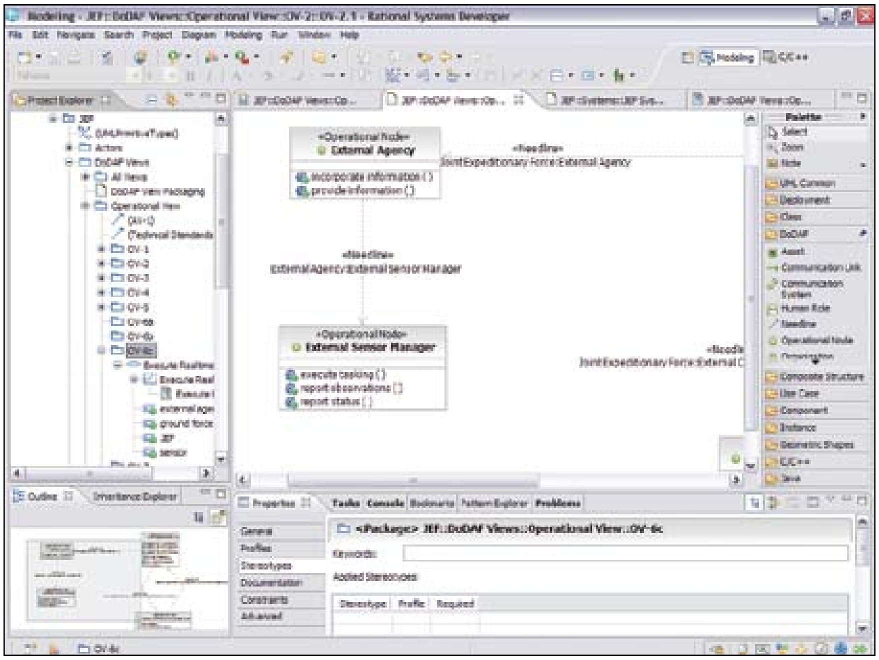


Figure 1: Build and manage DoDAF views and work products.

Model applications more productively than ever with UML 2

Rational Systems Developer software, based on the Unified Modeling Language, Version 2.1.1 (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, including freeform diagram support, aids in design, discovery and documentation activities. The UML 2 modeling capabilities in Rational Systems Developer software 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 capabilities. Support for the de facto Department of Defense Architecture Framework (DoDAF) standard is now conveniently included in the product as well.

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 system structure and behavior, 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 the underlying implementation technology.

The Object Management Group's (OMG's) answers to these needs are found in UML 2 and in 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. Rational Systems Developer software supports MDA 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.

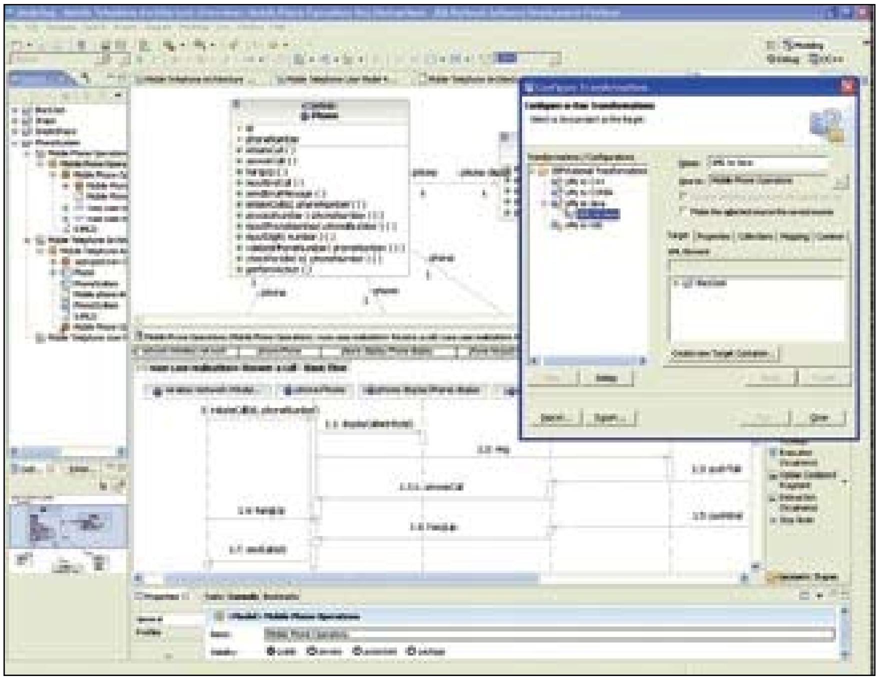


Figure 2. Rational Systems Developer software provides several usability features that make modeling and model-driven development more productive than ever.

Modeling for teams

With the Rational Systems Developer product, model files can be deconstructed into smaller components and placed under version control at a more granular level to better support parallel development activities. Its sophisticated model-merging capabilities let you compare, merge and reconcile models in an ongoing parallel development process. Even more sophisticated model-fusing capabilities let you leverage existing model components to create new logical model structures.

Tap into the power of model-based automations

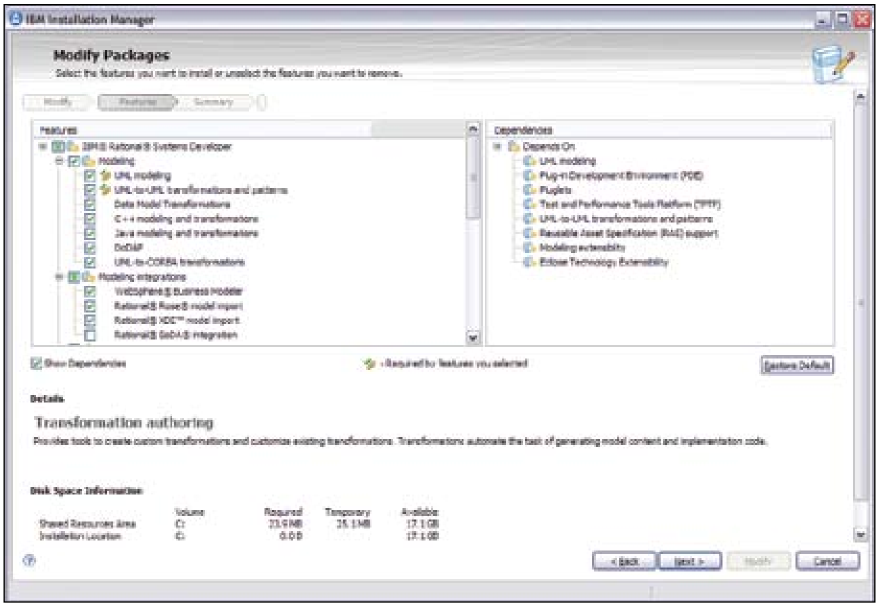
Modeling alone adds value and helps reduce project risk, but even greater benefits are realized when models are used to automate the creation of other development artifacts, including other models, code and more. Use the design patterns included with Rational Systems Developer software to help you more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

Rational Systems Developer software includes UML-based transformations for generating Java, C++ and CORBA languages. A forthcoming product will extend the transformation capabilities to include UML to C#, and C# to UML; it will also make it possible to visualize the contents (CTS types) of Microsoft® Visual Studio 2005 solutions and incorporate them into Rational Systems Developer diagrams. Rational Systems Developer software 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 Rational Systems Developer software includes tools for developing custom transformations that target any type of implementation outputs and transformations between UML models at different levels of abstraction.

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

Preserve Java architectures with structural review and control

Users can take advantage of structural review and control features for Java code to help preserve well-architected Java applications. When projects begin with existing code, designers can leverage structural review features to understand the system's structure before proceeding with design and development — so they are better able to modify the system without introducing undesired side effects (e.g., functionality, scalability and maintainability problems that may result from unexpected dependencies). Users can visualize existing design patterns and automatically detect structural antipatterns, which are undesired yet common software problems that can make software difficult to maintain and update.



- IBM® Rational® Systems Developer
  - ☑ Modeling
    - ☑ UML modeling
    - ☑ UML-to-UML transformations and patterns
    - ☑ Data Model Transformations
    - ☑ C++ modeling and transformations
    - ☑ Java modeling and transformations
    - ☑ DoDAF
    - ☑ UML-to-CORBA transformations
  - Modeling integrations
    - ☑ WebSphere® Business Modeler
    - ☑ Rational® Rose® model import
    - ☑ Rational® XDE™ model import
    - □ Rational® SoDA® integration

- Depends On
  - UML modeling
  - Plug-in Development Environment (PDE)
  - Plugins
  - Test and Performance Tools Platform (TPTP)
  - UML-to-UML transformations and patterns
  - Reusable Asset Specification (RAS) support
  - Modeling extensibility
  - Eclipse Technology Extensibility

☑ Show Dependencies

= Required by features you selected

Transformation authoring

Provides tools to create custom transformations and customize existing transformations. Transformations automate the task of generating model content and implementation code.

	Volume	Required	Temporary	Available
Shared Resources Area	C:	23.9 MB	25.1 MB	17.1 GB
Installation Location	C:	0.0 B		17.1 GB

Figure 3: The installation process allows users to install Rational tools into existing Eclipse instances, and to also install optional features.

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 the Eclipse project, all of the benefits of open software have come to the development environment itself. Because Rational Systems Developer software is built on top of the open Eclipse 3.2 platform, you can more easily extend the Rational software's features to meet specific project requirements through an ecosystem of third-party plug-ins. Eclipse is written in Java code, which means you can outfit your team across both Microsoft Windows® and Linux® desktops. You can leverage a Ready for IBM Rational Software partner plug-in, for example, to add SysML modeling or model-based simulation capabilities to the environment.

Because the features of the Rational Systems Developer software and the Eclipse platform are modular, installation is highly configurable. Install both the Eclipse environment and the Rational Systems Developer product. Or install just selected Rational Systems Developer 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, including the ability to share the same Eclipse environment with Wind River Workbench for a complete device development environment.

Integrate with the IBM Rational Software Delivery Platform

Integrated design and development improves traceability between code and its most immediately related artifacts. But complex software projects need traceability throughout the lifecycle. When requirements change, architects need to know which part of the architecture the change affects. 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 software processes. Failing to address these issues introduces increased risk to overall project success.

Rational Systems Developer software integrates with other components of the IBM Rational Software Delivery Platform. Use the Rational Systems Developer software to access requirements stored and managed in IBM Rational RequisitePro® software. Associate requirements to corresponding modeling elements and synchronize them with user-selectable rules. Generate reports highlighting traceability from requirements to design. Submit and track defects, run complex queries and access your "to do" list through the IBM Rational ClearQuest® integration. Use IBM Rational ClearCase® or IBM Rational ClearCase LT software—IBM's robust family of software configuration management products—to manage model files. Alternatively, Rational Systems Developer software integrates with Concurrent Versions System (CVS) for clients 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, including guidance specifically for system engineering. Included content for RUP systems engineering also provides specific process guidance and tool mentors, following a Model Driven Systems Development approach. These integrations help reduce risk and make application development more predictable. Leverage many of these convenient integrations through the IBM Rational Team Unifying Platform™ solution.

For more information

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

ibm.com/rational/adc

IBM Corporation
Software Group
Route 100
Somers, NY 10589
U.S.A.

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

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.