

TP731A 09/02

Rational. software



IBM Rational ClearCase Integration with IBM WebSphere Studio

Table of Contents

Introduction	1
Rational ClearCase and IBM WebSphere Studio Application Developer Integration Overview	1
Rational ClearCase Integration with IBM VisualAge for Java	2
Rational ClearCase: Market-Leading Software Configuration Management	2
About Rational ClearCase LT	2
About Rational ClearCase	3
Upgrading from Rational ClearCase LT to Rational ClearCase	4
The IBM WebSphere Studio Product Family	4
Getting Started: The Rational ClearCase Integration With IBM WebSphere Studio Application Developer	6
Integration Architecture	6
Installing Rational ClearCase in the Integrated IBM WebSphere Studio Application Developer Environment	6
Setting Rational ClearCase Preferences	6
Connecting to Rational ClearCase in the Application Developer Environment	7
Integrated Rational ClearCase and IBM WebSphere Studio Application Developer Environment Overview	7
Rational ClearCase Functions in the Application Developer Environment	9
Rational ClearCase Menu Options	9
Rational ClearCase ToolBar Icons	9
The Rational ClearCase Explorer	10
Creating Projects and Views in the Integrated Environment	10
What Files Should be Under ClearCase Source Control?	10
Creating a New Application Developer Project Under Rational ClearCase Control	10
Controlling an Existing Application Developer Project with Rational ClearCase	11
Adding an Existing Application Developer Project Under Rational ClearCase Control to the Application Developer Workspace	11
Adding Multiple Application Developer Projects to the Application Developer Workspace	11
Migrating Existing VisualAge for Java and Other Projects Already Under Rational ClearCase Control to Application Developer	11
Automated Migration of VisualAge for Java Projects Under Envy Source Control to Application Developer with Rational ClearCase	12
Migrating Application Developer 4.x Projects Already Under Rational ClearCase Control to Application Developer 5.x	13
Creating a New View	13
Creating Multiple Project Views	13
Updating Views in the Integrated Environment	14
Working on a Project in the Integrated Environment	14
Adding an Element to Source Control	14
Checkout and Edit	14
Save and Checkin	14
Closing Application Developer Before All Files are Checked In	15
Moving Files	15
Renaming Files	15
Deleting Files	15
Using Rational ClearCase with the IBM WebSphere Studio Application Developer Refactor Function	15
Performing Namespace Operations in Disconnected Mode	15
Using Rational ClearCase with the IBM WebSphere Studio Application Developer Compare/Replace History Function	15
Unified Change Management Operations in the Integrated Environment	16
Delivering From a Development Stream	16
Rebasing with Rational ClearCase in the Integrated Environment	16
Using the Rational ClearCase Build Function	16
Viewing Configuration Records in Application Developer	17
Accelerate Rational ClearCase Success with Rational Services	17
Glossary of Terms	19
Resources	20

Introduction

This white paper provides business and technical overviews of the integration between the IBM Rational ClearCase® products and IBM® WebSphere® Studio product family. Although the integration is applicable to all of the IBM WebSphere Studio product family, the remainder of this document is based on IBM WebSphere Studio Application Developer 5.0. It covers the benefits of the Rational ClearCase product family, the software configuration management functions available within IBM WebSphere Studio Application Developer (Application Developer), and the use of the Rational ClearCase product with WebSphere Studio Application Developer. Technologists will also benefit from installation information and usage scenarios.

IBM Rational ClearCase and IBM WebSphere Application Developer Integration Overview

IBM introduced IBM WebSphere Studio Application Developer, the core J2EE development environment for your e-business applications. It helps multi-functional development teams speed the deployment of complex e-business applications that encompass code, content, and components such as XML, HTML, Java™ servlets, Web Services and more. IBM Rational ClearCase is the market-leading software configuration management solution that scales from small workgroups to large enterprise development organizations. Whatever your software configuration management needs, there is a Rational ClearCase solution that can help you improve predictability and accuracy.

IBM Rational ClearCase has been integrated with IBM WebSphere Studio Application Developer. Much more than a high-level facility for checkin and checkout, this deep integration enables a wide array of Rational ClearCase software configuration management operations to be performed right from within the Application Developer environment. Benefits of this powerful integrated environment include the ability to:

- Speed development
- Quickly get up and running with the market-leading software configuration management solution
- Ensure software accuracy and team effectiveness
- Reduce deployment risk
- Lower training costs through a single familiar environment

For your convenience, the entry-level Rational ClearCase LT product is included in the following IBM WebSphere Studio product family configurations:

- WebSphere Studio Application Developer
- WebSphere Studio Application Developer Integration Edition
- WebSphere Studio Enterprise Developer

The Rational ClearCase product for medium and large-scale enterprise application development is not included on the Application Developer CD, however it works equally as well with the Rational ClearCase – Application Developer integration software. New and existing Rational ClearCase customers can take advantage of the same strong integration to ensure development accuracy.

IBM Rational ClearCase Integration with IBM VisualAge for Java

Integration also exists between IBM VisualAge® for Java integrated development environment and Rational ClearCase. This allows you to use the same software configuration management system—Rational ClearCase—across a development environment that encompasses both IBM VisualAge for Java and WebSphere Studio Application Developer. This white paper discusses the migration process for those customers who are presently using VisualAge for Java with Rational ClearCase and who wish to migrate to IBM WebSphere Studio Application Developer.

IBM Rational ClearCase: Market-Leading Software Configuration Management

Rational ClearCase is the market-leading software configuration management solution that simplifies the process of change. Rational ClearCase provides a family of products that scale from small project workgroups to the distributed global enterprise. It helps software teams control everything that evolves in software development—requirements, design models, source code, change requests, test scripts, and more.

Rational ClearCase products offer tools and processes that you can implement today and tailor as you grow without changing configuration management systems—no retraining, retooling or migrating data. By automating many necessary yet error-prone tasks, the Rational ClearCase product family enables teams to build the highest quality software in Internet time. On a single project, you can choose to use either Rational ClearCase LT or Rational ClearCase as the software configuration management solution.

About IBM Rational ClearCase LT

Rational ClearCase LT is the entry-level, scalable software configuration management solution that is ideally suited to small teams and small project workgroups who develop applications locally on a single server. Rational ClearCase LT, Microsoft® Windows® version, is conveniently included in IBM WebSphere Studio Application Developer packaging to provide easy access to software configuration management functions. Rational ClearCase LT for UNIX offers the same functionality and runs on the IBM AIX®, Red Hat Linux and other UNIX® operating systems. The Rational ClearCase LT, Red Hat Linux version, is also included in the WebSphere Studio Application Developer packaging.

Easy to install and manage, Rational ClearCase LT provides:

- *Powerful version control capabilities*—including the ability to version any file system artifact produced across the software development lifecycle, as well as directories that may evolve as the project progresses. It maintains complete, annotated histories of source code, binaries, executables, documentation, test suites, libraries and Web elements. Rational ClearCase LT employs a secure, read-only access model, where write access to files in the shared Versioned Object Base (VOB) is provided only when files have been checked out. A Versioned Object Base or VOB is the term used to describe the ClearCase configuration control storage repository.
- *Snapshot views*—provide easy access to the right versions of files and directories required to complete a specific task, while shielding team members from potentially destabilizing changes being made by other developers.
- *Deep integration with IBM WebSphere Studio Application Developer*—permits users to efficiently perform most Rational ClearCase functions without leaving the Application Developer user interface.
- *Parallel development support*—to dramatically accelerate development, Rational ClearCase LT supports parallel development activities. Features include automatic branching to enable multiple development teams to work independently from the same code base, and patented *diff/merge* technology that efficiently merges source code, HTML, and XML files.

- *Web interface for remote online access*—provides universal data access from standard Web browsers. From the browser, users can access views comprised of the “public” VOB and private work areas. Note that the version of ClearCase LT included with Application Developer does not include a license for the ClearCase Web interface. Therefore, if you wish to use the ClearCase Web interface, you must purchase either a full ClearCase or ClearCase LT license from IBM Rational.
- *Tight integration with Rational ClearQuest*— Rational ClearQuest® is the change and defect tracking solution that supports multiple levels of integration with Rational ClearCase. Rational ClearQuest can be used to relate change requests to a code base stored in Rational ClearCase LT and can provide prioritized “to do” lists. By using Rational’s Unified Change Management process, you can also take advantage of a more structured integration and workflow between these tools.
- *Process configurability*—enables you to transparently enforce configuration management policies across a project without dictating specific procedures that everyone must follow. Rational ClearCase provides customizable event triggers, object locks, granular access control, automatic branch creation and automatic e-mail notification that can all be used to guide development processes.
- *Unified Change Management (UCM)*—UCM is an optional, out-of-the-box, activity-based process for managing change. UCM can be selectively used on a project-by-project basis. Through Rational ClearCase and Rational ClearQuest, UCM helps managers prioritize and coordinate the many development activities that must be performed on hundreds of different artifact versions. With UCM, team members know what activities must be performed on which artifacts. UCM helps teams effectively “baseline” requirements together with code, models, test assets and more.
- *Easy installation to jumpstart development*— Rational ClearCase LT is an easy-to-install solution for a single-server architecture. Intuitive start-up wizards automate server set up, while built-in TCP/IP networking simplifies administration in a mixed (UNIX and Windows) environment.
- *Strong scalability and seamless upgrade path*—as your team’s needs change and grow, you may require enterprise-level software configuration management capabilities, such as advanced build management, data replication and multiple distributed servers. You can easily upgrade to Rational ClearCase while you preserve established VOBs and views. Rational ClearCase seamlessly meets your evolving development needs without re-training, re-tooling or changing processes.
- *Integrated with Rational Suite lifecycle development solution*— Rational ClearCase LT is included in Rational Suite®, the end-to-end integrated lifecycle development solution. As part of the powerful Rational Suite Team Unifying Platform, it enables development teams to improve development speed and software quality.

About IBM Rational ClearCase

Rational ClearCase is the industry-leading software configuration management solution that supports medium and large-sized development teams on a wider variety of platforms. You can easily upgrade to the comprehensive Rational ClearCase solution from Rational ClearCase LT as your team grows and your development projects become more complex. If your development teams perform enterprise-level software development, Rational ClearCase provides all of the functions included in Rational ClearCase LT plus:

- *Advanced build management*—enables you to streamline the build process while ensuring reproducibility and improving accuracy. The enterprise-level build management capabilities in Rational ClearCase optimize build times, guarantee the reproducibility of software versions, and audit builds. The Rational ClearCase *clearmake* and *omake* functions ensure accuracy during the build process by automatically detecting dependencies, reusing derived objects, and producing detailed build documentation. Rational ClearCase is compatible with internally-developed build programs and Windows/UNIX *make* programs, enabling existing applications to be placed under Rational ClearCase control.
- *Distributed Development Support*—enterprise development organizations can be comprised of geographically distributed teams creating complex applications in a parallel development environment on multiple distributed servers. Through the Rational ClearCase MultiSite® optional add-on, distributed organizations can effectively collaborate through locally-available, automatically-replicated VOBs.

- *Dynamic and snapshot views*—Two different types of views provide flexible workspace management. Available only in Rational ClearCase, dynamic views provide global, transparent access to any version of any element and are ideal for an audited build environment or a complex project that requires instantaneous updates to evolving source files. Ideally suited to disconnected users and early-stage projects that require isolation, snapshot views provide easy access to the right versions of artifacts.
- *Deployment integration with IBM WebSphere Application Server*—Available only for WebSphere servers, the Deployment Wizard for IBM WebSphere automates the deployment of files from IBM WebSphere Studio to IBM VisualAge for Java test environments, saving time and producing a deployment audit trail that is reproducible. The Deployment Wizard also automates transfers of IBM WebSphere Studio and IBM VisualAge for Java artifacts to a staging or production repository, providing a secure and reliable way of archiving and restoring an existing WebSphere application. This integration supports WebSphere Application Server, VisualAge for Java, and WebSphere Studio 3.5 and later.
- *IBM Mainframe Development Support*—Rational ClearCase for the IBM Mainframe supports both the IBM S/390® and IBM zSeries™ platforms running the SuSE Linux operating system. Rational ClearCase for the IBM Mainframe enables development organizations to achieve “economies of scale” by centralizing the administration, management and storage of development assets on a single powerful platform. The Rational ClearCase comprehensive platform coverage also offers organizations the ability to share artifacts between users working across the Windows, UNIX and Linux environments.

Upgrading from Rational ClearCase LT to Rational ClearCase

As your needs and your organization grow, the Rational ClearCase product family is unique in its ability to grow with you. When you have Rational ClearCase LT, it’s easy to upgrade to Rational ClearCase without retraining, retooling or migrating data. Rational ClearCase is an ideal solution when your needs include:

- Large or distributed project teams
- Reproducibility and auditability of software build versions
- Instantaneous updates to project elements
- Multiple development servers
- Transparent enforcement of configuration management policies
- Development environment support on the IBM zSeries mainframe platform
- Automated deployment options for IBM WebSphere and the VisualAge for Java test environment

To upgrade to Rational ClearCase, contact IBM Rational using one of the contact options listed at the end of this document.

The IBM WebSphere Studio Product Family

Initially e-business applications were HTML-based presentations of information. No longer. Today’s applications are more complex, leveraging HTML pages, XML components, a wide array of Java applications, servlets, EJB™ components, Web services, and more. Development tools that are vertically focused on either Web site development or Java application development are inadequate for rapidly integrating and maintaining complex applications. Additionally, you need development tools that have the flexibility to help you bridge between new J2EE applications and existing assets spanning across the enterprise. IBM WebSphere Studio, available in multiple configurations, is designed to meet these challenges.



Figure 1: The new IBM WebSphere Studio product family

IBM WebSphere Studio Workbench—an IBM branded and supported offering of Eclipse Framework. This is the foundation on which WebSphere Studio product family and business partner products are built on. This technology allows IBM and other tool vendors to participate in an open, simple, extensible, team-enabled, and portable development environment by creating “plug-ins”. WebSphere Studio Workbench is not a purchasable IBM product, but rather a technology that is seamlessly available in “plug-in” products. Optimized for the WebSphere software platform, it provides the base for accessing common services.

IBM WebSphere Studio Site Developer—the onramp to the WebSphere Studio, is an easy to use tool for developing and managing Web sites that include HTML, JSPs, Java, servlets, rich media, XML, and Web services. Bringing all aspects of application development into a common interface, it helps diverse Web application development teams access all of the information they need. This is the follow-on product for IBM WebSphere Studio Professional and WebSphere Studio Advanced version 4.

IBM WebSphere Studio Application Developer—the core J2EE™ development environment, helps you to optimize and simplify J2EE development and integration with best practices, templates, code generation for server-side (EJB and servlet) and client-side (HTML and JSP) development, deployment. It provides relational database tools for creating and manipulating relational database schemas and includes application performance profiling tools to help you detect application performance issues early in the development cycle to ensure quality. Available in two configurations, the Integration Edition adds visual tools for creating application workflows, industry leading support for a services-oriented architecture, and simplified connectivity to over 30 backend systems. This new product helps solve the most demanding e-business application development challenges. It is a single tool for developing and managing HTML, JSPs, Java, servlets, rich media, XML, JCA adapters, workflows, and Web services.

IBM WebSphere Studio Enterprise Developer—a Web and traditional development environment for the enterprise, this products brings the power of J2EE, rapid application development and team support to diverse enterprise application organization. This product includes all the functions of WebSphere Studio Application Developer, plus support for ASM, COBOL, PL/I, EGL (Enterprise Generation Language), Struts, visual assembly tools, and z/OS development tools. This is the follow-on product for VisualAge Enterprise Suite and VisualAge Generator.

IBM VisualAge for Java—is IBM’s integrated development environment for Java application development. WebSphere Studio Application Developer is the follow-on toolset that addresses extended e-business development needs. Included in the packaging for Application Developer during 2001 and 2002, VisualAge for Java can be used with Application Developer to complete projects that require development in both tools. Over time, functionality in VisualAge for Java will be migrated to Application Developer.

Getting Started: The IBM Rational ClearCase Integration With IBM WebSphere Studio Application Developer

This section discusses the integration architecture between Rational ClearCase and IBM WebSphere Studio Application Developer. It explains the installation process and includes a “guided tour” of the integrated environment workspace. Unless otherwise noted, Rational ClearCase is used in this and remaining sections of this document when discussing Rational ClearCase and Rational ClearCase LT capabilities.

Integration Architecture

As illustrated in Figure 2, IBM WebSphere Studio Application Developer is a plug-in for the IBM WebSphere Studio Workbench. The Rational ClearCase integration software, which works equally well with Rational ClearCase and Rational ClearCase LT, is also an extension to WebSphere Studio Workbench. When other tools are plugged into the architecture, Rational ClearCase operations can be performed on other artifacts produced by these other tools.

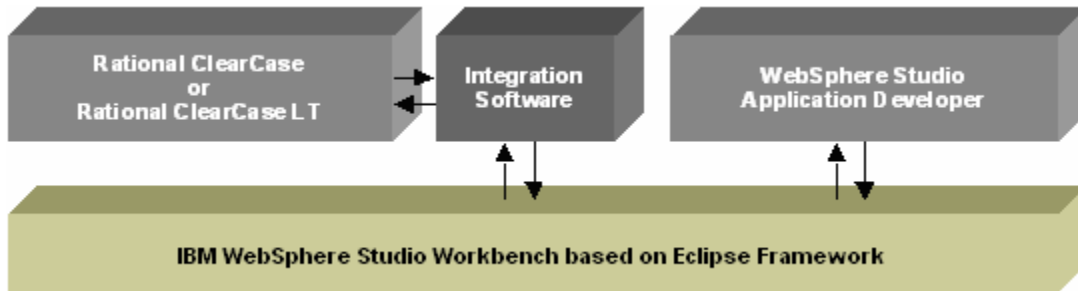


Figure 2: Rational ClearCase – IBM WebSphere Studio Application Developer integration architecture.

Installing Rational ClearCase in the Integrated IBM WebSphere Studio Application Developer Environment

The Application Developer product CD includes the Rational ClearCase integration software component as well as the Rational ClearCase LT product. The installation process for this integrated environment has been designed to flexibly meet the needs of several different customer use scenarios.

The Rational ClearCase Integration with WebSphere Studio Application Developer is an extension component that is optionally installed during the Application Developer installation process. This process does not install the Rational ClearCase LT product. You can choose to install Rational ClearCase LT from the main Application Developer Installation Launcher or you can install ClearCase LT directly from the Application Developer CD. If you are planning to use Rational ClearCase instead of Rational ClearCase LT in this environment and do not already have it installed, you must purchase and install it.

Setting Rational ClearCase Preferences

You can change Rational ClearCase preferences and configuration options within the Application Developer user interface to suit your development needs. To access Rational ClearCase preferences, select *Window > Preferences* from the Application Developer pull-down menu. The Preferences dialog box appears, which enables you to configure both Application Developer and Rational ClearCase functions. Click on *Team -> Rational ClearCase* to set configuration options.

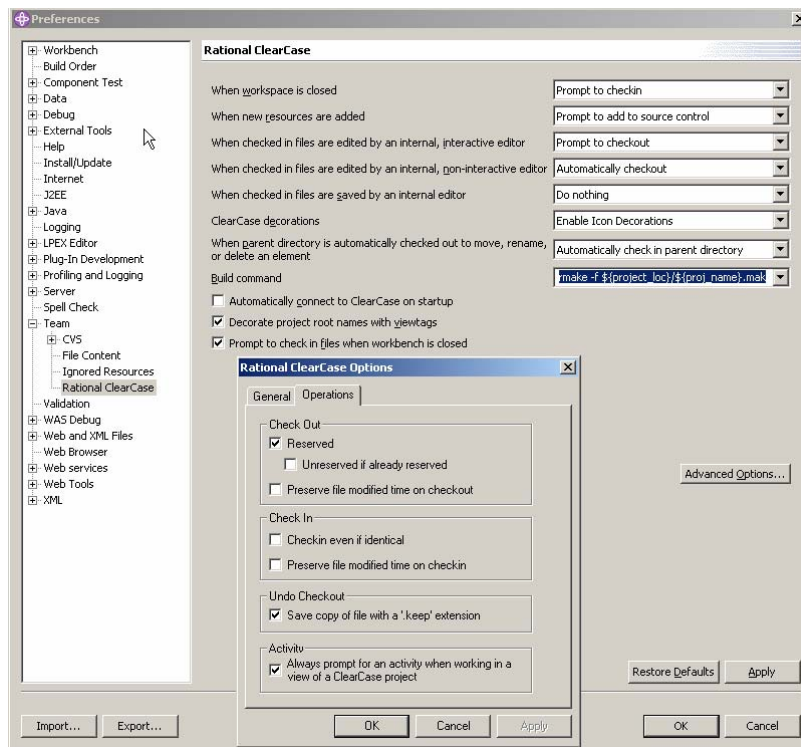


Figure 3: IBM WebSphere Studio Application Developer and Rational ClearCase integrated Preferences dialog box.

Connecting to Rational ClearCase in the Application Developer Environment

Rational ClearCase allows you to work both online and offline. You must connect to Rational ClearCase within the Application Developer environment before you can begin using those Rational ClearCase functions that require access to Rational ClearCase servers. Examples include delivering completed code and rebasing your stream before performing specific activities. Click the *Connect to Rational ClearCase* button on the Rational ClearCase toolbar before you perform functions that require access to Rational ClearCase servers. Any resources under Rational ClearCase control that are open in the Application Developer Navigator window are refreshed as a background operation so that you can continue working while the refresh occurs. It is also possible to automatically connect to ClearCase upon startup of Application Developer. To do this, select the *Automatically connect to ClearCase on startup* option in the ClearCase preferences (see *Setting Rational ClearCase Preferences* for details).

Integrated Rational ClearCase and IBM WebSphere Studio Application Developer Environment Overview

This section provides a “guided tour” of the integrated user interface. The integrated user interface incorporates common functions provided by WebSphere Studio Workbench, as well as Application Developer and Rational ClearCase functions. For the most part, you will notice that it is an easy-to-use, streamlined environment. You cannot discern whether a function is derived from the WebSphere Studio Workbench or from Application Developer. You will also notice that Rational ClearCase functions are easy to access, yet are often discernable.

The term *workspace* has different meanings in Application Developer and in Rational ClearCase. In Application Developer, a workspace is the set of projects a user can work on in that instance of Application Developer. In Rational ClearCase a workspace generally means a Rational ClearCase view. A view provides controlled, versioned access to the exact set of files or file versions in the VOB. Rational ClearCase views simplify the work environment by providing the elements necessary to complete particular tasks while shielding the complexity of the larger development environment. Remember that in Rational ClearCase LT, you can create snapshot views, while in Rational ClearCase you can create both snapshot and dynamic views. An Application Developer workspace may not necessarily incorporate everything found in a Rational ClearCase view, for example, it may not include documentation files stored in ClearCase that are not part of an Application Developer project. Additionally, an Application Developer workspace may also include items that are not under ClearCase control, for example, it may include a project folder that is not under Rational ClearCase control. For purposes of clarity

in this white paper, the term “Application Developer workspace” is used and a Rational ClearCase view or workspace is simply called a “view”. Also note that an Application Developer view has a different meaning than that of a ClearCase view. An Application Developer view is a type of window within the Application Developer interface.

See Figure 4. The labels indicate different work areas and functions in the integrated environment.

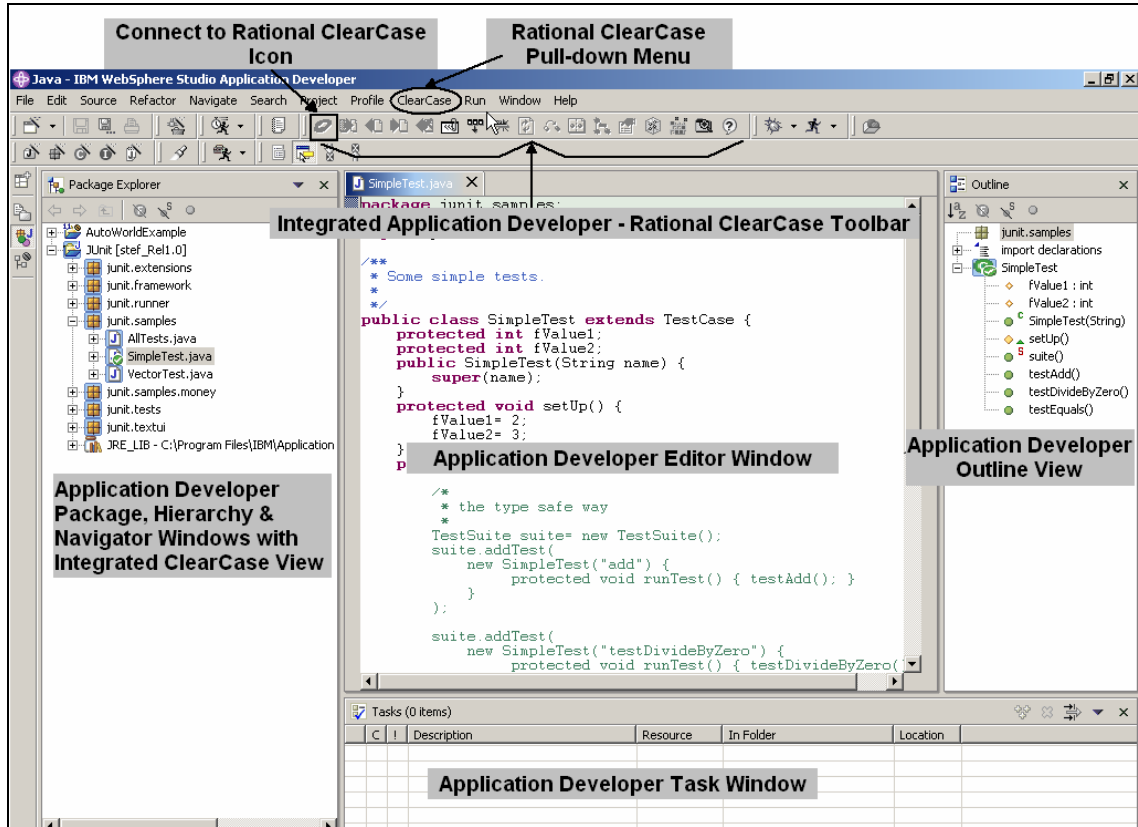


Figure 4: Screen shot illustrating the integrated WebSphere Studio Application Developer – Rational ClearCase environment. All functions are tightly integrated in a single environment to provide ease of use and development speed.

- *Connect to Rational ClearCase*— Rational ClearCase is off-line in Application Developer until you click on the *Connect to Rational ClearCase* button. This connects you to Rational ClearCase server(s).
- *Rational ClearCase Pull-down Menu*—enables you to perform specific Rational ClearCase operations right from within the Application Developer workspace. When you are using Rational ClearCase LT, some items on this menu will be grayed out because these options are only available in the Rational ClearCase product. To enable these operations, upgrade to Rational ClearCase. For a list of operations that are available on this pull-down menu, refer to the table below.
- *Application Developer and Rational ClearCase Toolbar*—This toolbar (or corresponding pull-down menu) is used to perform Application Developer and Rational ClearCase functions. You can quickly and easily perform Rational ClearCase operations and UCM functions through a simple mouse click right from within the Application Developer workspace. Source control operations are also available on common project files through the right mouse button and Team context menu, and the ClearCase pull-down menu.
- *Application Developer Package, Hierarchy and Navigator Window with Integrated Rational ClearCase View*—An Application Developer function, the Navigator window provides shared access to common files. You see Rational ClearCase integration in this area through a view that provides controlled,

versioned access to common projects stored within Rational ClearCase VOBs. Next to the directories and files in this window, you see icons that illustrate various Rational ClearCase version control states such as checked out, checked in, etc. You will notice that there are two projects in the window. One of them is under the control of Rational ClearCase and the other is not. The Package, Hierarchy and Navigator windows show different aspects of project elements. This white paper refers to the Navigator Window that is typically used by development team members to access project files.

- *Application Developer Editor Environment*—In the center is the WebSphere Studio Application Developer coding environment, where development tasks such as Java and EJB coding can be performed.
- *Application Developer Outline View*— Displays a visual interface to the contents of the editor. For example, it provides access to Java fields and methods of the currently selected Java class.
- *Application Developer Task Window*—Shows user-defined tasks, compilation errors, and validation errors related to Application Developer projects.

Rational ClearCase Functions in the Application Developer Environment

From within the Application Developer environment, you can perform many Rational ClearCase functions. As we have just seen, these options are accessed in one or more ways: through the Rational ClearCase pull-down menu; by selecting a file, right-clicking, and selecting the Team option from the context menu; or by clicking on icons on the ClearCase toolbar. The following table is an overview of Rational ClearCase functions that can be performed from within the Application Developer user interface. This white paper does not cover every function; however, you can refer to the Rational ClearCase online help within the Application Developer user interface for more information.

<i>Team Context Menu Options</i>	<i>Rational ClearCase Menu Options</i>	<i>Rational ClearCase ToolBar Icons</i>
Add to Source Control	Connect to Rational ClearCase	Connect to Rational ClearCase
Check Out	Set Current Activity	Add to Source Control
Check In	Deliver Stream	Check Out
Undo Check Out	Rebase Stream	Check In
Update	Update View	Undo Check Out
Refresh Status	ClearCase Build *	Set Current Activity
Hijack	Run ClearCase Explorer	Deliver Stream
Compare With Previous Version	Show Project Checkouts	Rebase Stream
	Run Merge Manager	Refresh Status
Show Properties	Create New View	Update
Show Version Tree	Edit View Properties	Compare With Previous Version
Show History	Start/Stop View *	Show Version Tree
	Mount/Unmount VOB *	Show Properties
	ClearCase Help	Update View
		ClearCase Build *
		Run ClearCase Explorer
		ClearCase Help

* Available with Rational ClearCase only

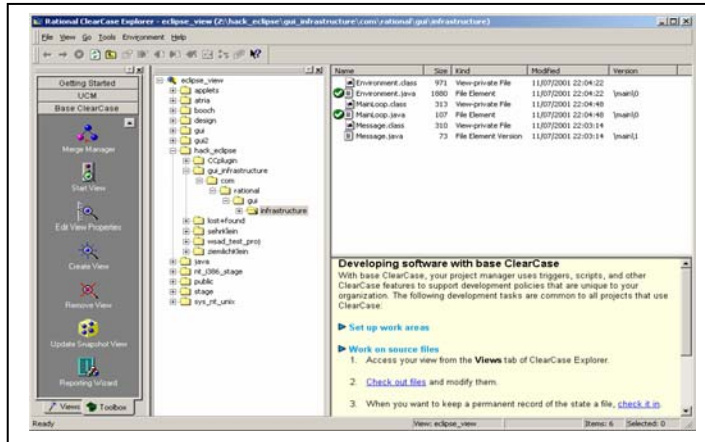


Figure 5: Rational ClearCase Explorer.

The Rational ClearCase Explorer

The Rational ClearCase Explorer is a graphical user interface that is separate from the Application Developer Navigator. It lets you manage and view software assets under Rational ClearCase control. This graphical user interface provides access to Rational ClearCase functions such as creating views, accessing meta data, accessing the reporting wizard and more. In Application Developer, you can conveniently access the Rational ClearCase Explorer from the ClearCase pull-down menu or through the ClearCase toolbar.

Creating Projects and Views in the Integrated Environment

You can create or join projects and create one or more views of VOBs with Rational ClearCase. This section discusses the process of establishing projects and views under different scenarios.

What Files Should be Under ClearCase Source Control?

In general, the following files should be version-controlled in ClearCase, either in a staging VOB for that purpose or in their development location:

- All files that are part of a software release
- All *.project* files
- All *.classpath* files
- All Project Set (*.psf*) files
- Tools and configuration files involved in the build process

Note: All *.project* files and *.classpath* files should be version-controlled in their development location. When ClearCase prompts you to add the *.project* file or *.classpath* file to source control, click *Yes*.

Creating a New Application Developer Project Under Rational ClearCase Control

To create a new version-controlled project, use the *File > New > Project* option on the Application Developer toolbar or menu. A *New Project* dialog box appears where you specify a project name. You must change the location of the project to a VOB location by unchecking the *Use Default option* and selecting a path in a Rational ClearCase view. Through a view, you are selecting a directory within a VOB to store project files. The *Add Elements to Source Control* dialog box appears. You choose those elements that you wish to add to source control in the specified VOB. Next you see that the project files you have placed under source control appear in the Navigator with version control status icons next to them. See Figure 6.

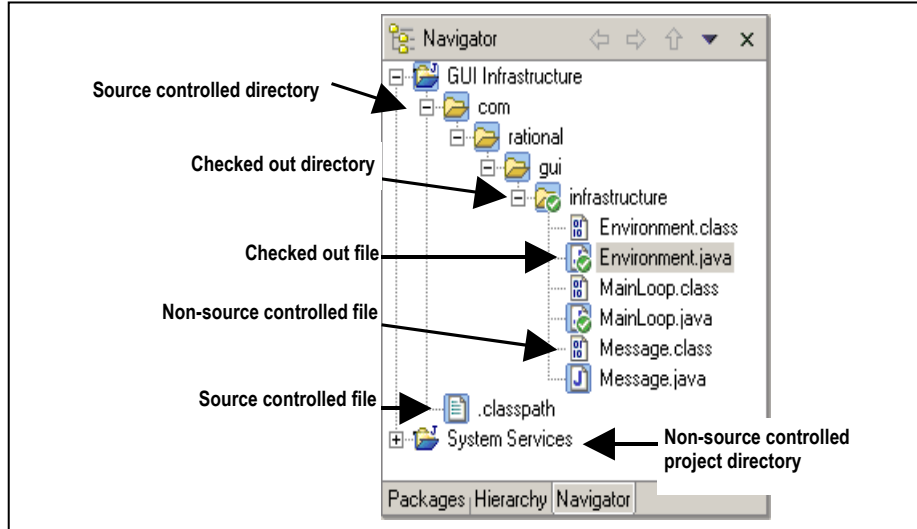


Figure 6: WebSphere Studio Application Developer projects under Rational ClearCase source control.

Controlling an Existing Application Developer Project with Rational ClearCase

An existing project that has already been worked on in the Application Developer workspace can be placed under the control of Rational ClearCase. This is done by selecting and right clicking on the project and selecting *Team > Share Project* option from the context menu and then selecting Rational ClearCase from the list of repository types. You are then prompted to enter a directory path within a Rational ClearCase VOB. Use this option if you are moving a WebSphere Studio project into a Rational ClearCase VOB. Do not use the *Move* option within Application Developer for this purpose. Rational ClearCase source control icons appear next to the project directories and files in the Navigator. See Figure 6.

Adding an Existing Application Developer Project Under Rational ClearCase Control to the Application Developer Workspace

Existing Application Developer projects may be stored in a VOB but not be in the Application Developer workspace yet. This is typically the case when a user first installs Application Developer, begins working on a multi-person project, or begins work on a new version of a project. These projects can remain under the control of Rational ClearCase and can be added to the newly created Application Developer workspace. To do this, select *File > Import* from the pull-down menu in Application Developer and select *Existing Project into Workspace*. The Application Developer *Import Project* dialog box appears, where you specify the directory location for the existing project in the Rational ClearCase VOB. It is important to enter the directory location that is the root directory of the project. The project should now appear in the Application Developer Navigator window.

Adding Multiple Application Developer Projects to the Application Developer Workspace

It is also possible to import multiple projects into a workspace at one time by using Project Sets. To use project sets, you must first create a Project Set file (.psf) by exporting the list of projects in an Application Developer workspace. Select *File > Export* from the Application Developer menu and then select *Team Project Set* to create the project set file. It is recommended that the project set file be stored in a ClearCase VOB. Once the project set has been created, it can be imported into other workspaces by selecting *File > Import* from the Application Developer pull-down menu and then selecting *Team Project Set*. Once the *Team Project Set* dialog appears, select the location in the VOB where the Project Set file (.psf) exists. Note that it is possible for project sets to contain some projects that are under source control and others that are not under source control.

Migrating Existing VisualAge for Java and Other Projects Already Under Rational ClearCase Control to Application Developer

Existing projects that were developed with IBM VisualAge for Java or other tools and that are already under the control of Rational ClearCase can be added to Application Developer. This process is similar to creating a new project, however you must ensure that project characteristics are carefully set in the *New Project* dialog box to match those of your existing project.

To do this, use the *File > New > Project* option on the Application Developer toolbar or menu. A *New Project* wizard appears. You must specify the same project name and project type, and ensure that other characteristics of your project match the existing project. Uncheck the *Use Default Location* box and select the directory in a Rational ClearCase view in which the project resides. The project directories and files appear in the Navigator and Package windows with Rational ClearCase source control icons next to them. See Figure 6.

Refer to the IBM documentation for additional information and considerations on migrating projects that were created in VisualAge for Java or other tools into Application Developer, especially for EJB applications.

Automated Migration of VisualAge for Java Projects Under Envy Source Control to Application Developer with Rational ClearCase

If you use the Visual Age for Java Envy source code repository to store projects and artifacts, you can choose to automatically migrate to Rational ClearCase in Application Developer by using a Rational partner product called CodePro™ Migrator for Rational® ClearCase®. Instantiations, a Rational Technology Partner and an advanced IBM Business Partner, provides this automated code repository migration tool. Now you can easily migrate existing applications that are under Envy source control to Application Developer with Rational ClearCase.

CodePro Migrator for Rational ClearCase converts and migrates:

- Source code
- Resources
- Structures
- Traceability attributes — ownership, timestamps and derivation history

to Rational ClearCase in Application Developer. Solutions, projects, and packages are fully converted to ClearCase structures.

Prior to performing the migration you perform initial analysis to identify target VisualAge for Java assets and determine how the components are related. CodePro Migrator provides dependency information to aid in your analysis. CodePro Migrator supports several approaches to performing the migration:

- *Map VisualAge for Java Projects to Application Developer Projects with Rational ClearCase* - You can analyze your VisualAge for Java projects and determine if the projects represent individual programs or subsystems of a larger program. Individual programs map into separate Application Developer projects, and also into separate view directories. Depending on build dependencies, complex programs map into either a series of Application Developer projects, one for each subsystem, or a single Application Developer project. If subsystems are mapped to Application Developer projects, they must be constructed with appropriate dependencies. Complex programs depending on other Application Developer projects require a separate directory for each project.
- *Migration of Current and Historical Versions of Projects to Rational ClearCase* - You select your solutions, projects, or packages to migrate, and a view directory that will contain the migrated code and artifacts. You can create views and VOBs using Rational ClearCase wizards, or select an existing view. Filtering mechanisms aid in the selection of a limited set of versions. The mapping of items to Rational ClearCase is collected into an export set. The export set and the mapping parameters are stored persistently in VisualAge for Java Envy source control. The export-import is performed according to the export set, and successful migrations are recorded.
- *Update Rational ClearCase Based on New Development Done in IBM VisualAge for Java* - You can update your Rational ClearCase VOB by re-invoking the migration of an export set. New items and new versions of existing items can be migrated to Rational ClearCase. Older versions can be added as branches in the derivation tree. In the case of a migration conflict, in which you may have selected a version number that already exists in Rational ClearCase, you can re-map the version number and run the migration again.

- *Re-target the Migration* - If you decide to re-target your export set to a different view directory, the portions of the export set that support update tracking can be flushed so migration and update can be tracked in the new directory.

There are some special considerations for exporting EJB-based applications from the VisualAge for Java environment into the Application Developer environment. For more information, refer to IBM WebSphere Studio Application Developer *readme* file and look for section 5.0 on migration which provides sources of more information. For more information on Instantiations CodePro Migrator for Rational ClearCase, please visit www.instantiations.com/codepro/migrator

Migrating Application Developer 4.x Projects Already Under Rational ClearCase Control to Application Developer 5.x

In Application Developer 4.x, there was no notion of “self-describing projects” which maintain the properties of a project in the project directory. Therefore, it was necessary for the ClearCase integration to implement this notion using a *.cc_meta file*. Since Application Developer 5.0 now creates self-describing projects using the *.project* file, it is necessary to migrate a project under ClearCase control from Application Developer 4.x to Application Developer 5.x.

To do this, you will need to import the project by selecting the *File > Import > Existing WebSphere Studio 4.x ClearCase Project into Workspace* menu option. If the project has never been imported into Application Developer 5.x before, this will import the project with the project properties from the ‘*.cc_meta*’ file. These properties will be stored in a new ‘*.project*’ file. It is recommended that you select the option to *Delete obsolete WebSphere 4.x meta file* in the *Import* dialog. Attempts to import a migrated project again using this operation are not allowed.

Creating a New View

A Rational ClearCase view provides access to the exact versions of files required to complete particular tasks while shielding complexity. Views can be created directly from within the integrated Application Developer environment. To create a new view, use the *ClearCase > Create New View* menu option. You are presented with a *View Creation Wizard* where you are prompted for various options of your view, including UCM and development stream options. Note that creating a view does not make new elements visible in the Application Developer user interface. You have to create or import projects from a VOB (through the view) into the Application Developer workspace before any elements become visible in Application Developer. See *Adding an Existing Application Developer Project Under Rational ClearCase Control to the Application Developer Workspace* for details.

Source files are generally versioned and shared with other project team members, while build files are typically private and not shared. Users must ensure that the right files are versioned, i.e., a user may not want to version and share build files and build settings, but may wish to version and share source files. Application Developer provides an option for ignoring specific file types that you do not wish to add to source control. The Rational ClearCase integration adds a default list of file types to this list of *Ignored Resources*. Refer to the Application Developer help on *Ignored Resources* for more information.

Creating Multiple Project Views

Often, developers work with multiple views of the same project, and these views must be accessible from the Application Developer workspace. **It is strongly recommended that you associate one Rational ClearCase view with a unique Application Developer workspace.** For example, if you want to work in a parallel development environment where you have a development branch for bug fixes for release A and new development for release B, you would have two different Application Developer workspaces, one for View A and one for View B.

After a new view is created, you must associate a workspace with it. To do this, you can either create a new shortcut to Application Developer that specifies the workspace directory, or start Application Developer from the command line and specify the appropriate workspace directory. It is helpful to use the same name as the corresponding Rational ClearCase view. When you start Application Developer from the command line, use the *-data* flag and the workspace storage directory path or create a shortcut with the *-data* flag and path. The workspace storage directory will be created automatically if it does not already exist. The following example demonstrates starting Application Developer using a workspace associated with a view called ‘*alex_Rel3*’.


```
C:\> "C:\Program Files\IBM\Application Developer\wsappdev.exe" -data  
C:\workspaces\alex_Rel3
```

When you create or import a project into the Application Developer workspace, you must ensure that you use the correct view when specifying the project location. Refer to the online help for more information.

Updating Views in the Integrated Environment

Rational ClearCase in the Application Developer workspace can be used to bring different versions of files into a project and to synchronize the files in the Application Developer workspace with the Rational ClearCase VOB. The *Update View* function in Rational ClearCase updates the user's snapshot view. It finds and automatically refreshes checkout status information, new files and versions in the project's VOB, and files that have been edited while in disconnected mode. Any changes resulting from the Update View command will automatically be reflected in the Navigator window.

When changes have been made to the Application Developer workspace file system outside of Application Developer (e.g., using an external text editor), Application Developer does not automatically synchronize with the Rational ClearCase VOB. When this occurs you must use the Application Developer *Refresh* function to cause changes to appear in Application Developer.

To change the configuration specification (config spec) or load rules of a view, use the Rational ClearCase *Edit View Properties* dialog box available from the ClearCase menu bar within Application Developer.

Working on a Project in the Integrated Environment

As you work on a project, you may wish to perform a variety of change management functions. This section describes these functions.

Adding an Element to Source Control

Within the Application Developer environment, you may specify that new project objects or elements be placed under the control of Rational ClearCase. To do this, right-click on the file and choose *Team > Add to Source Control*. The *Add Element to Source Control* dialog box appears and you can place the element under Rational ClearCase control. If you are using UCM, you are also prompted to specify an activity to associate with the new element creation.

Checkout and Edit

In order to edit a file stored in a Rational ClearCase VOB, it first must be checked out. To save time and speed development, Rational ClearCase checkout functions can be performed right from the Navigator window in the Application Developer workspace. To check out a single or multiple files, select the files and choose the Rational ClearCase checkout function by right-clicking on the object to access the Team context menu, or by selecting the checkout icon on the ClearCase tool bar. The Rational ClearCase checkout dialog box appears and additional comments may be provided. If you are using UCM, you are also prompted to specify an activity which describes the changes that have been made in plain language. This information is helpful in gaining a quick understanding of how the code has changed and in selecting files to perform future Rational ClearCase operations, such as *Deliver Stream*. Once the operation is complete, the Rational ClearCase check mark icon appears next to the checked out file in the Navigator window. The developer can then edit the file as necessary.

It is also possible to enforce the checkout of files the first time they are edited; either via an *interactive* or *non-interactive* edit. An *interactive* edit requires input from a user, for example editing a Java file. A *non-interactive* edit does not involve any input from the user and is usually initiated by some other operation such as, a refactor operation. To enforce checkouts on edits for either case, open the Application Developer Preferences Window, select *Team > Rational ClearCase* and then specify *Prompt for Checkout* or *Automatically checkout* for the field *When checked in files are edited by an internal, interactive editor*. Select *Automatically checkout* for the field *When checked in files are edited by an internal non-interactive editor*.

Save and Checkin

When editing is complete, save the file. In the Navigator window, select the file that has just been saved and choose the checkin option through the right-click context menu, on the ClearCase pull-down menu, or on the ClearCase toolbar. The checkin dialog box appears so that the selected file can be checked in and additional comments may be provided. If you are using UCM, the activity used at the time of checkout will be used for checkin. When the operation is complete, the Rational ClearCase checkout icon disappears from the file in the Navigator window.

Closing Application Developer Before All Files are Checked In

When you are attempting to close Application Developer, Rational ClearCase brings to your attention any files that remain checked out. Rational ClearCase notes those files that are still checked out and prompts you to check them in if desired. You may choose to leave them checked out so you can continue working on them next time you start Application Developer. It is also possible to configure the integration to automatically check in all files when the workspace is closed. This option can be configured in the Application Developer Preferences window under the *Team > Rational ClearCase* page.

Moving Files

When you use the Application Developer *Move* function and the files are under the control of Rational ClearCase, the file locations in the Rational ClearCase VOB will be automatically updated such that the ClearCase history of the file is maintained in the new location. This automatically reconciles the Application Developer workspace with the Rational ClearCase VOB.

Renaming Files

When you use the *Rename* function in the Application Developer workspace and the files are under the control of Rational ClearCase, the file names in the Rational ClearCase VOB will be automatically updated to match those in the Application Developer workspace and the ClearCase history for these files will be maintained.

Deleting Files

Deleting files from the Application Developer workspace has a similar impact on the Rational ClearCase VOB. The deleted files disappear from the Application Developer workspace, and the file names will be automatically removed from the corresponding Rational ClearCase VOB directory. Note that the entire element is not removed from the VOB, it is simply removed from the VOB directory's namespace using the ClearCase *rmname* operation.

Using Rational ClearCase with the IBM WebSphere Studio Application Developer Refactor Function

Refactor is an Application Developer function that enables changes to classes and methods to be automatically reflected in all affected application components. Two example cases are: changing a Java method name or signature, and changing a Java class name.

When *Refactor* changes a method name or signature, it also updates all source code that refers to the method. When this occurs, Rational ClearCase automatically checks out the affected files. The refactor changes are made and the files remain checked out. You must then manually observe the changes and check the files in.

When *Refactor* performs a change to a class name, the file is also renamed. This is the same process as renaming a file—the file names in the Rational ClearCase VOB will be automatically updated and reconciled with the class name changes in the Application Developer workspace. Dependent classes are handled in the manner described above in method name changes, i.e., files are checked out when modified and must be manually checked in.

If UCM is being employed on the project, an Activity ID is required for check outs and check ins. The activity that was associated with the file when it was automatically checked out during the refactor operation is presented at checkin time. If necessary, you may use the *Set Activity ID* function on the ClearCase toolbar to change the activity ID before performing the refactor operation, thus causing these checked out versions to be included in the changed activity ID.

Performing Namespace Operations in Disconnected Mode

In order to perform namespace operations, such as move, rename and delete on any files stored under ClearCase control in Application Developer you must first Connect to ClearCase. If you are working in a Snapshot view disconnected from the network and must perform a namespace operation, you will not be able to perform namespace operations within Application Developer.

Using Rational ClearCase with the IBM WebSphere Studio Application Developer Compare/Replace History Function

Application Developer has a *Compare and Replace Local History* operation, which is exposed in the Java IDE portion of Application Developer, the project Navigator window, the Package browser and the Outliner. This history mechanism applies to local changes performed in a user's workspace and enables you to go back to an earlier edit. The functionality that creates a new entry in the local history is invoked when a file is saved; it does not apply to merging multiple branches of changes. To perform a Compare or Replace of Local History operation, right mouse click on an element in the Navigator, Outline or Package Browser windows and select

either the *Compare With* or *Replace With* option. A dialog box is displayed to select the needed edit of the file. The file must be checked out before you are able to use the *Replace With* operation..

The Rational ClearCase History mechanism performs a different function—its history tracks the evolution of versions of files created globally across the project team. To view a Rational ClearCase history, use the Rational ClearCase *Version Tree Browser* or *History Browser* and merge tools to view version differences and merge across multiple file versions.

Unified Change Management Operations in the Integrated Environment

This section discusses the Rational ClearCase *Deliver* and *Rebase* functions that may be used on a development project that employs Unified Change Management or UCM. IBM Rational recommends having one Application Developer workspace associated with your Integration view and one workspace for each of your development views. See *Using Views in the Integrated Environment* for more information.

Delivering From a Development Stream

Deliver Stream is a Rational ClearCase function that enables you to deliver and integrate all or specific change sets to the integration stream once an activity has been completed. For example, if your task was to create new feature A, you may deliver a change set relating only to new feature A into the stream.

To do this, select the *ClearCase > Deliver Stream* function or the toolbar icon. A dialog box is presented. Select the development stream that you would like to deliver. Whenever you checked in a set of changes into Rational ClearCase, you specified the activities that you performed. Now, at integration time, you can choose which activities you would like to deliver into the integration stream. For example, your changes may have implemented new feature A, and now you can choose to deliver only changes relating to new feature A. You can also specify the stream that you would like to deliver the changes to. After delivery, Rational ClearCase enables you to examine the integration stream with the changes. When you are satisfied with the results of the merge in the integration view, check in the code by choosing *Complete* on the *Deliver Stream* dialog box.

As you deliver a stream, you will probably have two different instances of Application Developer open—one workspace will have the private development workspace which contains the complete changes, and the other will have the integration workspace open. When you invoke the *Deliver Stream* function, you will have modified the file system outside of Application Developer. In order to refresh the integration workspace with the results of the merge in the integration view, use the Application Developer *Refresh* function.

Rebasing with Rational ClearCase in the Integrated Environment

Rebase Stream updates your private Rational ClearCase development stream to include any changes made between the last baseline you were using and the baseline that you choose to use at this point. The baselines exist on the integration stream. When you perform the *Rebase Stream* operation, you are effectively merging changes from the integration stream into your private development stream. This operation merges changes in the new baseline with changes you may have made in your development stream but have not yet delivered to the integration stream. To perform this operation, choose the *ClearCase > Rebase Stream* menu option or click on the *Rebase Stream* icon on the ClearCase toolbar. The *Rebase Stream* function will not affect versions of files that are associated with any activities that are already underway but have not been delivered. Any file system changes, such as new files, resulting from the Rebase operation will automatically be updated within your workspace.

You must use the Application Developer *Refresh* function in order to see the specified baseline in the Application Developer workspace associated with your development view.

Using the Rational ClearCase Build Function

The enterprise-level build management capabilities in Rational ClearCase optimize build times, guarantee the reproducibility of software versions, and audit builds. The Rational ClearCase *clearmake* and *omake* tools ensure accuracy during the build process by automatically detecting dependencies, reusing derived objects, and producing detailed build documentation. Choose the *ClearCase > Build* option or the *Build* icon from the toolbar to begin the process of compiling your executable code according to the specifications in the *makefile*. The ClearCase build option, icon and menu option are available only in Rational ClearCase, not Rational ClearCase LT. Note that the project *makefile* must be in the root project directory before you perform a build operation. It is recommended that you place the *makefile*—which must be named *<project_name>.mak*—under source control.

It is possible to configure the ClearCase Build function in Application Developer to use either *clearmake* or *omake*. To do this, open the Application Developer Preferences window, select the *Team > Rational ClearCase* page and then specify your desired build command.

Invoking the Build function will simply launch the associated build command with the project makefile. You will need to check the status of the build operation outside Application Developer.

Viewing Configuration Records in Application Developer

Building applications with *omake* produces *derived objects (DO)*, which contain *configuration records* that contain detailed information on how the DOs were built. By default, this information is not displayed from within the Application Developer environment. However, it is possible to configure the External Tools mechanism to display configuration records within Application Developer. To do this, select *Run > External Tools > Configure* and then select *New* in the External Tools Configuration dialog. In the *New External Tool* dialog, enter *Display Config Record* for the External Tool name. For the *Tool Location* field, browse to the *bin* directory within the ClearCase installation directory and select *cleartool.exe*. For the *Tool Arguments* field, enter *"cater -recurse \${resource_loc}"*. Finally, ensure that the *Show execution log on console* is checked and then select OK. To display the configuration record for any class built with *clearmake*, select the *.class* file and then select *Run > External Tools > Display Config Record*. You should see the configuration record displayed in the Application Developer Log Console.

Accelerate Rational ClearCase Success with Rational Services

Rational provides world-wide services that get you up and running quickly, improve self-sufficiency and accelerate project implementation:

- With Rational Developer Networksm (RDN) at your fingertips, you gain a vast Web-based resource of technical expertise for quickly building your software development skills. RDN is an online community of experts, mentors and software professionals dedicated to sharing knowledge about software design, development and lifecycle maintenance.
- Rational University complements IBM's comprehensive suite of development tools by providing you with a unique role- and team-based curriculum that combines a proven software development process and best practices with a structured professional education curriculum. Both classroom instruction and on-site training options are available. You can take advantage of these Rational ClearCase courses: Rational ClearCase Basics, Rational ClearCase Fundamentals, Applying Rational ClearCase Meta Data, Rational ClearCase Administration, Rational ClearCase MultiSite Administration, Principles of Software Configuration Management, Introduction to Rational ClearCase UCM, Basics of Rational ClearCase Unified Change Management, Fundamentals of Rational ClearCase and Unified Change Management, and Managing Software Projects with Rational ClearCase and Unified Change Management. Rational University also offers a comprehensive list of web-based training courses available on RDN. See www.rational.net for more information.
- Rational Professional Services brings you qualified consultants with years of experience to help accelerate any phase of your development project. In addition to consulting and mentoring, Rational Professional Services offers customized services.
- Rational Worldwide Technical Support ensures that your project keeps moving forward with the help you need, when you need it. Our maintenance programs offer efficient problem resolution via phone, e-mail, Web and fax, as well as up-to-date access to Rational's latest software releases.

Glossary of Terms

Application Developer Workspace—the set of projects that the user can work on in that instance of Application Developer

Branch or Code Branch—in a parallel development environment, a branch or code branch is a development path. You can see the evolution of branches in a Rational ClearCase Version Tree Browser which graphically depicts different branches of code that have been created to enable parallel development over time. Branches isolate the work of developers whose changes should not yet be shared and enable developers to continue working while integration, build and test periods occur.

Stream – in UCM, a stream is a long-lived ClearCase object. It is a member of a single UCM project and is a mechanism for creating and recording configurations. A stream identifies the exact set of versions currently available for you to view, modify, or build.

Rational ClearCase View—provides access to exact versions of files in the Rational ClearCase Versioned Object Base (VOB); serves to shield the developer from the complexity of the larger project environment while completing a specific task. Snapshot views are available in Rational ClearCase and Rational ClearCase LT; Dynamic views are available in Rational ClearCase.

Rational ClearCase Explorer—interface that provides access to a wide range of Rational ClearCase functions. The Rational ClearCase Explorer is accessible from within the Application Developer user interface.

Diff/merge—A Rational ClearCase function, this patented technology allows you to visually compare two or more file versions, and merge them into a single file. Diff/merge is a critical function for enabling parallel development.

Plug-in—software that enables IBM or a third party tool to plug into WebSphere Studio Workbench. The benefit is the ability to assemble disparate tools into a common development environment that precisely meets development needs for complex e-business applications.

Refactor—an Application Developer function that enables automatic updates to dependent classes and methods across all affected application components. Refactor may involve moving or renaming classes and methods and this has source control implications.

Versioned Object Base (VOB)—a Versioned Object Base or VOB is the term used to describe the Rational ClearCase source control storage repository.

IBM WebSphere Studio Workbench—new IBM development tools platform based on the Eclipse Framework. Through plug-ins, IBM and other tools vendors can participate in an open, simple, extensible development environment. While not a purchasable product, common Workbench functions are seamlessly available in tools such as Application Developer.

Resources

Rational ClearCase

Product information: www.rational.com/products/clearcase/index.jsp

Technical Support Information Page: www.rational.com/support/products/clearcase.jsp

ClearCase Plugins: www.rational.com/support/downloadcenter/addins/index.jsp

Rational ClearQuest

Product information: www.rational.com/products/clearquest/index.jsp

Technical Support Information Page: www.rational.com/support/products/clearquest.jsp

Rational Technical Support

Phone: 800-433-5444

E-mail: support@rational.com

Internet support request form: www.rational.com/support/forms/supportrequest-new.jsp

Rational Developer Network (RDN)

www.rational.net

IBM WebSphere Studio Application Developer:

www.ibm.com/software/ad/studioappdev

www.ibm.com/websphere/developer/library/techarticles/0108_studio/studio_beta.html

IBM WebSphere Studio Workbench:

www.ibm.com/software/ad/workbench

www.developer.ibm.com/welcome/wstools/workbench.html

IBM WebSphere Software Platform Main Page:

www.ibm.com/websphere



IBM software integrated solutions

IBM Rational supports a wealth of other offerings from IBM software. IBM software solutions can give you the power to achieve your priority business and IT goals.

- *DB2[®] software helps you leverage information with solutions for data enablement, data management, and data distribution.*
- *Lotus[®] software helps your staff be productive with solutions for authoring, managing, communicating, and sharing knowledge.*
- *Tivoli[®] software helps you manage the technology that runs your e-business infrastructure.*
- *WebSphere[®] software helps you extend your existing business-critical processes to the Web.*
- *Rational[®] software helps you improve your software development capability with tools, services, and best practices.*

Rational software from IBM

Rational software from IBM helps organizations create business value by improving their software development capability. The Rational software development platform integrates software engineering best practices, tools, and services. With it, organizations thrive in an on demand world by being more responsive, resilient, and focused. Rational's standards-based, cross-platform solution helps software development teams create and extend business applications, embedded systems and software products. Ninety-eight of the Fortune 100 rely on Rational tools to build better software, faster. Additional information is available at www.rational.com and www.therationaledge.com, the monthly e-zine for the Rational community.

Rational is a wholly owned subsidiary of IBM Corp. (c) Copyright Rational Software Corporation, 2003. All rights reserved.

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

Printed in the United States of America
01-03 All Rights Reserved.
Made in the U.S.A.

IBM the IBM logo, DB2, Lotus, Tivoli and WebSphere are trademarks of International Business Machines Corporation in the United States, other countries, or both.

Rational, and the Rational Logo are trademarks or registered trademarks of Rational Software Corporation in the United States, other countries or both.

Microsoft and Windows NT are registered trademarks of Microsoft 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.

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States, other countries, or both.

UNIX is a trademark of The Open Group in the United States, other countries or both.

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

The IBM home page on the Internet can be found at ibm.com