

Using Unified Change Management with Rational Suite®

VERSION: 2001A.04.00

PART NUMBER: 800-024443-000

support@rational.com
<http://www.rational.com>

Rational®
the e-development company™

IMPORTANT NOTICE

COPYRIGHT

Copyright ©2000-2001, Rational Software Corporation. All rights reserved.

Portions Copyright ©2000-2001, Compaq Computer Corporation. All rights reserved.

Portions Copyright ©1992-2001, Summit Software, Inc. All rights reserved.

Part Number: **800-024443-000**

Version Number: **2001A.04.00**

PERMITTED USAGE

THIS DOCUMENT CONTAINS PROPRIETARY INFORMATION WHICH IS THE PROPERTY OF RATIONAL SOFTWARE CORPORATION ("RATIONAL") AND IS FURNISHED FOR THE SOLE PURPOSE OF THE OPERATION AND THE MAINTENANCE OF PRODUCTS OF RATIONAL. NO PART OF THIS PUBLICATION IS TO BE USED FOR ANY OTHER PURPOSE, AND IS NOT TO BE REPRODUCED, COPIED, ADAPTED, DISCLOSED, DISTRIBUTED, TRANSMITTED, STORED IN A RETRIEVAL SYSTEM OR TRANSLATED INTO ANY HUMAN OR COMPUTER LANGUAGE, IN ANY FORM, BY ANY MEANS, IN WHOLE OR IN PART, WITHOUT THE PRIOR EXPRESS WRITTEN CONSENT OF RATIONAL.

TRADEMARKS

Rational, Rational Software Corporation, Rational the e-development company, ClearCase, ClearCase Attache, ClearCase MultiSite, ClearDDTS, ClearQuest, ClearQuest MultiSite, DDTS, Object Testing, Object-Oriented Recording, ObjecTime & Design, Objectory, PerformanceStudio, ProjectConsole, PureCoverage, PureDDTS, PureLink, Purify, Purify'd, Quantify, Rational, Rational Apex, Rational CRC, Rational Rose, Rational Suite, Rational Summit, Rational Visual Test, Requisite, RequisitePro, RUP, SiteCheck, SoDA, TestFactory, TestFoundation, TestMate, The Rational Watch, AnalystStudio, ClearGuide, ClearTrack, Connexis, e-Development Accelerators, ObjecTime, Rational Dashboard, Rational PerformanceArchitect, Rational Process Workbench, Rational Suite AnalystStudio, Rational Suite ContentStudio, Rational Suite Enterprise, Rational Suite ManagerStudio, Rational Unified Process, SiteLoad, TestStudio, VADS, among others, are either trademarks or registered trademarks of Rational Software Corporation in the United States and/or in other countries. All other names are used for identification purposes only, and are trademarks or registered trademarks of their respective companies.

Microsoft, the Microsoft logo, Active Accessibility, Active Client, Active Desktop, Active Directory, ActiveMovie, Active Platform, ActiveStore, ActiveSync, ActiveX, Ask Maxwell, Authenticode, AutoSum, BackOffice, the BackOffice logo, bCentral, BizTalk, Bookshelf, ClearType, CodeView, DataTips, Developer Studio, Direct3D, DirectAnimation, DirectDraw, DirectInput, DirectX, DirectXJ, DoubleSpace, DriveSpace, FrontPage, Funstone, Genuine Microsoft Products logo, IntelliEye, the IntelliEye logo, IntelliMirror, IntelliSense, J/Direct, JScript, LineShare, Liquid Motion, Mapbase, MapManager, MapPoint, MapVision, Microsoft Agent logo, the Microsoft eMbedded Visual Tools logo, the Microsoft Internet Explorer logo, the Microsoft Office Compatible logo, Microsoft Press, the Microsoft Press logo, Microsoft QuickBasic, MS-DOS, MSDN, NetMeeting, NetShow, the Office logo, Outlook, PhotoDraw, PivotChart, PivotTable, PowerPoint, QuickAssembler, QuickShelf, RelayOne, Rushmore, SharePoint, SourceSafe, TipWizard, V-Chat, VideoFlash, Virtual Basic, the Virtual Basic logo, Visual C++, Visual C#, Visual FoxPro, Visual InterDev, Visual J++, Visual SourceSafe, Visual Studio, the Visual Studio logo, Vizact, WebBot, WebPIP, Win32, Win32s, Win64, Windows, the Windows CE logo, the Windows logo, Windows NT, the Windows Start logo, and XENIX, among others, are either trademarks or registered trademarks of Microsoft Corporation in the United States and/or in other countries.

Sun, Sun Microsystems, the Sun Logo, Ultra, AnswerBook 2, medialib, OpenBoot, Solaris, Java, Java 3D, ShowMe TV, SunForum, SunVTS, SunFDDI, StarOffice, and SunPCi, among others, are trademarks or registered trademarks of Sun Microsystems, Inc. in the U.S. and other countries.

FLEXIm and GLOBEtrotter are trademarks or registered trademarks of GLOBEtrotter Software, Inc. Licensee shall not incorporate any GLOBEtrotter software (FLEXIm libraries and utilities) into any product or application the primary purpose of which is software license management.

BasicScript is a registered trademark of Summit Software, Inc.

Portions of this product incorporate the expat XML parser 1.0 under the Mozilla 1.1 license available at <http://www.mozilla.org/MPL/MPL-1.1.txt>. The source code version of the expat XML parser is available at <http://www.jclark.com/xml/expat.html>.

Apache Software Foundation Notice

The Tomcat software, which is distributed with Rational RequisiteWeb, is protected by the Apache Software License, Version 1.1. Copyright © 1999 The Apache Software Foundation. All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The end-user documentation included with the redistribution, if any, must include the following acknowledgement: "This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>)." Alternately, this acknowledgement may appear in the software itself, if and wherever such third-party acknowledgements normally appear.
4. The names "The Jakarta Project", "Tomcat", and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
5. Products derived from this software may not be called "Apache" nor may "Apache" appear in their names without prior written permission of the Apache Group.

THIS SOFTWARE IS PROVIDED "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation. For more information on the Apache Software Foundation, please see <http://www.apache.org/>.

PATENT

Portions covered by U.S. Patent Nos. 5,193,180 and 5,335,334 and 5,535,329 and 5,835,701 and 5,574,898 and 5,649,200 and 5,675,802.

U.S. Patents Pending.

International Patents Pending.

Purify is licensed under Sun Microsystems, Inc., U.S. Patent No. 5,404,499.

GOVERNMENT RIGHTS LEGEND

Use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in the applicable Rational Software Corporation license agreement and as provided in DFARS 277.7202-1(a) and 277.7202-3(a) (1995), DFARS 252.227-7013(c)(1)(ii) (Oct. 1988), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 227-14, as applicable.

WARRANTY DISCLAIMER

This document and its associated software may be used as stated in the underlying license agreement. Rational Software Corporation expressly disclaims all other warranties, express or implied, with respect to the media and software product and its documentation, including without limitation, the warranties of merchantability or fitness for a particular purpose or arising from a course of dealing, usage, or trade practice.

Contents

Preface	ix
Audience	ix
Other Resources	ix
Rational Suite Documentation Roadmap	x
Contacting Rational Technical Support	xi
1 What Is UCM?	13
The Building Blocks	13
Why Use UCM?	14
Typical Way to Use UCM	14
A Note About Tools	16
For More Information	16
2 UCM Concepts	17
UCM Basics	17
Project	17
Component	17
Activity	17
Stream	17
Baseline	18
ClearCase Basics	18
VOBs and Their Contents	18
Views and Streams	20
Using ClearCase	20
Behind the Scenes	20
ClearQuest Basics	20
For More Information	21
3 Configuring UCM: Planning and Setting Up Servers	23
Overview of Configuration	23
Planning a Network Environment	24
Setting Up Vendor Databases	25

Setting Up a ClearCase LT Server	25
Installing and Configuring the ClearCase LT Server.	26
Starting the ClearCase LT Server Setup Wizard	26
Working with the Server Setup Wizard.	26
For More Information.	28
4 Configuring UCM: Setting Up the Project Manager’s Computer	29
Installing Rational Suite.	29
Configuring ClearQuest	29
Setting Up a Schema Repository	30
Setting Up a User Database	31
Guidelines for Choosing a Schema	31
Creating a User Database	31
Adding Users to the User Database.	31
Optional: Configuring the User Database	32
Configuring ClearCase LT.	32
Setting UCM Policies	33
Joining the Project	34
Configuring Streams and Creating Views.	34
Sharing the Integration View Directories	35
Creating Additional Integration Views	36
Adding Another Component to the Project.	37
Creating the Component	37
Adding the Component’s Baseline to the Integration Stream.	37
Updating Development and Integration Views	38
Creating and Setting Up a Rational Project	39
Creating a Rational Project.	39
Creating a Share Name	39
Creating the Rational Project	40
Configuring a Rational Project	41
Associating Requirements	41
Associating Test Assets	43
Associating Change Requests.	44
Associating Rose Models.	45
Creating a Baseline	47
Updating Integration Views.	47
Configuring the RequisitePro / ClearQuest Integration	47
For More Information.	48

5	Configuring UCM: Setting Up Client Computers	49
	Installing Rational Suite	49
	Registering the Rational Project	49
	Joining the UCM Project	50
	Registering a Rational Project	51
	Establishing a Virtual Path Map in Rose	52
	Establishing a ClearQuest Connection	52
6	Developing Software	53
	Developing Requirements and Test Assets	53
	Developing Rose Visual Models and Code	53
	Starting the ClearCase Explorer	54
	Joining a Project	54
	Working On an Activity	55
	Setting an Activity: With ClearQuest	55
	Setting an Activity: Without ClearQuest	56
	Checking Out Files	56
	Changing Files	56
	Checking In Files	57
	Delivering an Activity	57
	Preparing Work Areas	57
	Starting the Deliver Operation	58
	Merging	58
	Inspecting and Testing	59
	Completing the Deliver Operation	59
	Your Project Manager Creates Baselines	59
	Rebasing Your Development Stream	60
	For More Information	61
7	Working with Baselines	63
	Checking In All Assets	63
	Displaying the Configure Project Dialog Box	64
	Checking In Requirements	64
	Checking In Test Assets	65
	Starting the ClearCase Project Explorer	65
	Locking the Integration Stream	65
	Creating the Baseline	66
	Unlocking the Integration Stream	66
	Building and Testing the Software Components	66

Promoting the Baseline and Working with Promotion Levels	67
Promoting All Baselines to the Recommended Level.	67
Changing One Component's Baseline Promotion Level.	67
Declaring the Recommended Promotion Level	68
Changing the List of Promotion Levels.	68
Creating a New RequisitePro Project or Test Datastore from a Baseline	69
For More Information.	69
8 Variations in Your Environment	71
Using UCM With or Without ClearQuest.	71
Using UCM With ClearCase Instead of ClearCase LT	72
Using ClearCase or ClearCase LT Without Using UCM	73
For More Information.	73
Index	75

Preface

This manual shows you how to set up and use Unified Change Management (UCM) with Rational Suite:

- UCM is Rational Software's approach to managing change in software system development from requirements to release. It is a change management process layered on top of Rational ClearCase and Rational ClearQuest.
- Rational Suite delivers a comprehensive set of integrated tools that embody software engineering best practices and span the entire software development lifecycle. Rational Suite's unparalleled level of integration improves communication both within teams and across team boundaries, reducing development time and improving software quality.

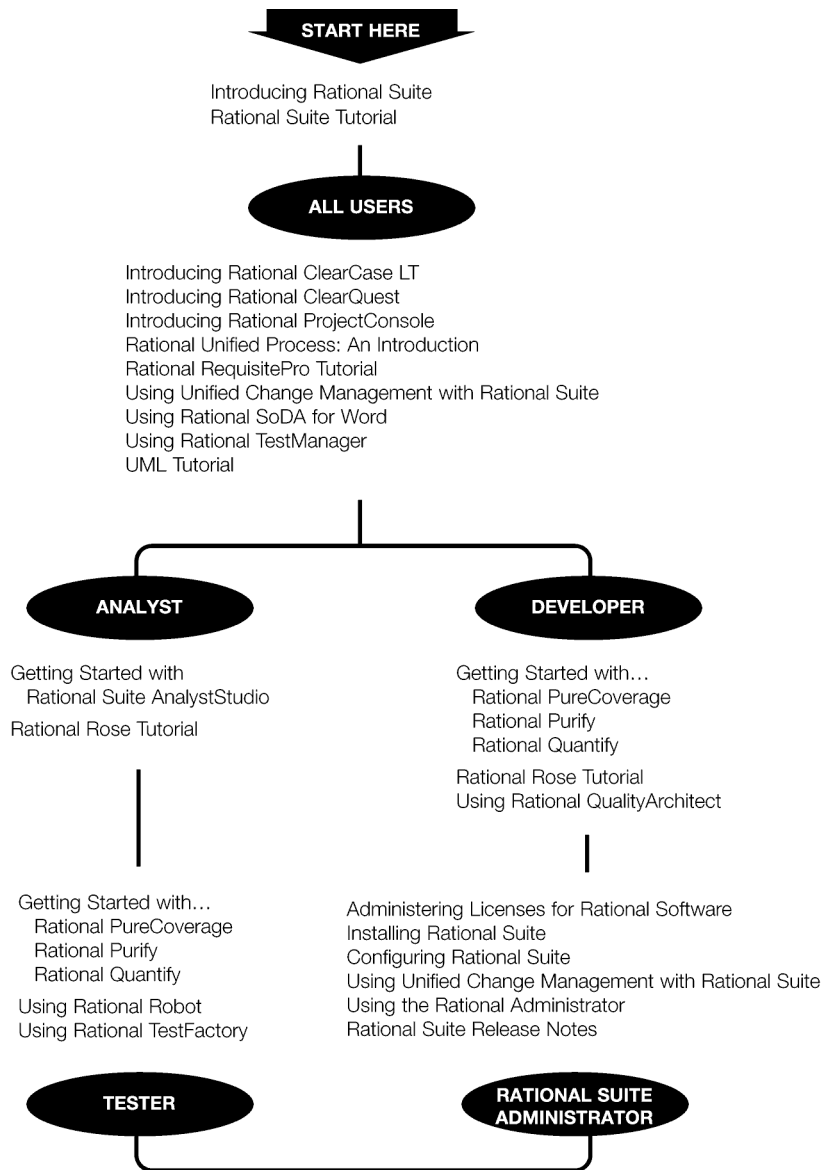
Audience

This manual is intended for administrators, project managers, and all members of the software development team, including requirements developers, software architects and developers, and quality engineers.

Other Resources

- All manuals are available online, either in HTML or PDF format. The online manuals are on the Rational Solutions for Windows Online Documentation CD.
- To send feedback about documentation for Rational products, please send e-mail to techpubs@rational.com.
- For more information about Rational Software technical publications, see: <http://www.rational.com/documentation>.
- For more information on training opportunities, see the Rational University Web site: <http://www.rational.com/university>.
- For articles, discussion forums, and Web-based training courses on developing software with Rational Suite products, join the Rational Developer Network by selecting **Start > Rational Suite > Logon to the Rational Developer Network**.

Rational Suite Documentation Roadmap



Contacting Rational Technical Support

If you have questions about installing, using, or maintaining this product, contact Rational Technical Support as follows:

Your Location	Telephone	Facsimile	E-mail
North America	(800) 433-5444 (toll free) (408) 863-4000 Cupertino, CA	(781) 676-2460 Lexington, MA	support@rational.com
Europe, Middle East, Africa	+31 (0) 20-4546-200 Netherlands	+31 (0) 20-4545-201 Netherlands	support@europe.rational.com
Asia Pacific	+61-2-9419-0111 Australia	+61-2-9419-0123 Australia	support@apac.rational.com

Note: When you contact Rational Technical Support, please be prepared to supply the following information:

- Your name, company name, telephone number, and e-mail address
- Your operating system, version number, and any service packs or patches you have applied
- Product name and release number
- Your case ID number (if you are following up on a previously reported problem)

Unified Change Management (UCM) is Rational Software's approach to managing change in software system development from requirements to release. UCM spans the development lifecycle, defining how to manage change to requirements, design models, documentation, components, test cases, and source code.

This manual helps you understand how to use UCM effectively to manage change in your software development environment.

The Building Blocks

UCM focuses on these concepts:

- An *activity* is a piece of work to accomplish in order to advance the state of your project. An activity might arise from an issue raised in a meeting, a defect entered into a defect database, or a request for enhancement sent in by a customer. Activities are tied to product features and requirements.
- An *artifact* is an item, usually a file, that you want to store under version control. At a more conceptual level, artifacts can be requirements, tests, visual models, code, project plans, and so on.

One of UCM's key strengths is that it links the activities used to plan and track project progress with the artifacts that are undergoing change. UCM is supported by both process and tools:

- The Configuration and Change Management workflow in the Rational Unified Process describes the process of using UCM.
- Rational ClearQuest manages a project's activities – the tasks, defects, and requests for enhancement – and provides charting and reporting tools that can help you track project progress.
- Rational ClearCase manages all artifacts produced by a software project.

Why Use UCM?

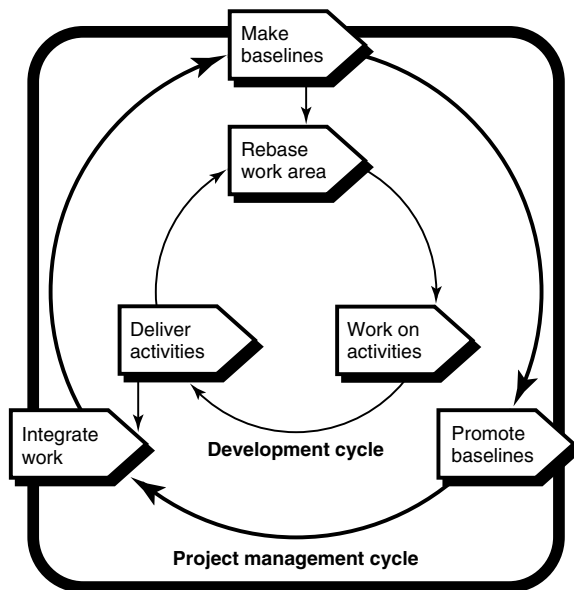
Rational customers and Rational employees have successfully used ClearCase and ClearQuest for years. During that time, common patterns have emerged regarding how users set up and use the tools. Based on our experience and observations, we've created UCM, a process that helps you get started with ClearCase and ClearQuest quickly and easily.

UCM raises the level of abstraction of working with these software configuration management and change management tools. For example, in a typical software configuration environment, you need to track the individual files to change. By contrast, with UCM, you work on activities, letting the tools keep track of the details for you (for example, which files changed).

Typical Way to Use UCM

It can be intimidating to consider adding more process to your work day. UCM is designed to automate much of the bookkeeping for you so that you need to track fewer details. Figure 1 and the following steps describe a typical way to use UCM:

Figure 1 Typical UCM Workflow



- 1 [one time] A project manager installs Rational software. The project manager sets up the ClearCase environment and creates a Rational project. (A *Rational project* associates Rational tools and enables integrations among them.) For more information, see *Configuring UCM: Planning and Setting Up Servers* on page 23.
- 2 [one time] You identify yourself to the project by *joining the project*. As a result, a private workspace (consisting of a *development stream* and a *development view*) is created for you. You also gain access to a workspace available to your entire team. This public workspace consists of an *integration stream*; you can create a companion *integration view* for your own use.
- 3 Your project manager uses ClearQuest to assign activities to you.
- 4 You run a ClearQuest query to find the activities assigned to you. This is your to-do list. From this list, you decide which activity to work on.
- 5 You work with artifacts as usual. Using ClearCase, you work in your private development workspace to:
 - Check out artifacts. When you check out an artifact, ClearCase asks which activity you want to work on. In the background, ClearCase keeps track of the *change set* (the list of changed artifacts) associated with the activity.
 - Edit and verify the changes.
 - Check the artifacts back in. When you check in an artifact, it is still part of your private workspace. Your change does not become publicly available until you deliver it, as described in Step 6.
- 6 When you have finished work on the activity, you *deliver* changes for the entire activity. Because ClearCase keeps track of the change set, you don't have to specify the list of artifacts to deliver. Delivering the changes makes the changes publicly available through the integration stream (for more information, see *Developing Software* on page 53). It may also close the activity, depending on the policies you establish.
- 7 After developers deliver a set of activities, the project manager creates a *baseline*, a new common starting place for all developers that includes the new activities. On your project, a new baseline may be created on a regular basis, perhaps even daily. For more information, see *Working with Baselines* on page 63.
- 8 If the changes in the baseline are approved (through testing or through another review process), your project manager promotes it, making it the recommended baseline.

- 9 You *rebase* your development stream so that when you work on your next activity, you're starting from the most recent recommended baseline. Restart with Step 4 to select the next activity to work on.

The remainder of this manual describes these steps in more detail and describes other scenarios you may follow when using UCM.

A Note About Tools

In this manual, we assume that you are using ClearCase LT and ClearQuest. ClearCase LT is a simplified ClearCase product and is included with every edition of Rational Suite.

Your environment may differ from the one described in this manual. For example, your group might use ClearCase (rather than ClearCase LT), or your UCM environment might not include ClearQuest. You might use ClearCase or ClearCase LT without using UCM. *Variations in Your Environment* on page 71 discusses these alternative environments. However, even if you work in a different environment, most of the information in this manual is still relevant.

In this manual, we discuss the UCM tools – Rational ClearCase and Rational ClearQuest – and the following Rational Suite tools that are integrated with UCM:

- Rational RequisitePro, which helps you create and manage requirements.
- Rational Rose, which helps you create visual models of software architecture and design.
- Rational Test, which helps you create, manage, and run automated tests.

For More Information

This chapter provides a starting point for understanding UCM. For more information, refer to the following resources:

- To learn more about the Rational Unified Process, read *The Rational Unified Process: An Introduction* by Philippe Kruchten. Addison-Wesley, 2000.
- To learn more about UCM in general, read *Software Configuration Management Strategies and Rational ClearCase: A Practical Introduction* by Brian White. Addison-Wesley, 2000.
- To learn more about how project managers can set up and manage a UCM environment, read *Managing Software Projects with ClearCase (Windows Edition)*, part of the ClearCase documentation set.

The chapter, *What Is UCM?* on page 13, describes how you typically use UCM. This chapter helps you understand terms and concepts used in UCM, ClearCase, and ClearQuest.

UCM Basics

UCM combines the power of ClearCase and ClearQuest to provide a complete, out-of-the-box, activity-based change management process.

Project

UCM organizes development work into *projects*. The project includes policies that govern how developers get access to and update the artifacts (typically, the files and directories) used in their development effort.

Component

Each project has one or more *components*, which are groupings of related sets of files. For example, you might have one component for each of these parts of your project: the GUI, the data definitions, and the business logic.

Activity

An *activity* is a piece of work to accomplish in order to advance the state of your project. In UCM, it is an object that tracks the work required to complete a development task.

Stream

A *stream* helps define your workspace. A project has one integration stream, which is part of the public workspace, and multiple development streams, each of which is part of a developer's private workspace. To complete each workspace, you need a *view*, described on page 20.

You typically work with a development stream, then deliver your work to the integration stream. The development stream tracks the activities assigned to you and lets you work in isolation from the rest of the project team.

Baseline

A *baseline* is a collection of some or all of the activities delivered to date. A baseline provides a common starting point for all developers on your project: you can *rebase* your development stream so that it contains the changes in the most recent recommended baseline.

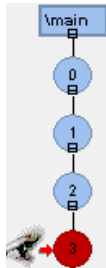
ClearCase Basics

ClearCase helps you manage your project's artifacts – the files that change as a result of moving your project closer to its next release.

VOBs and Their Contents

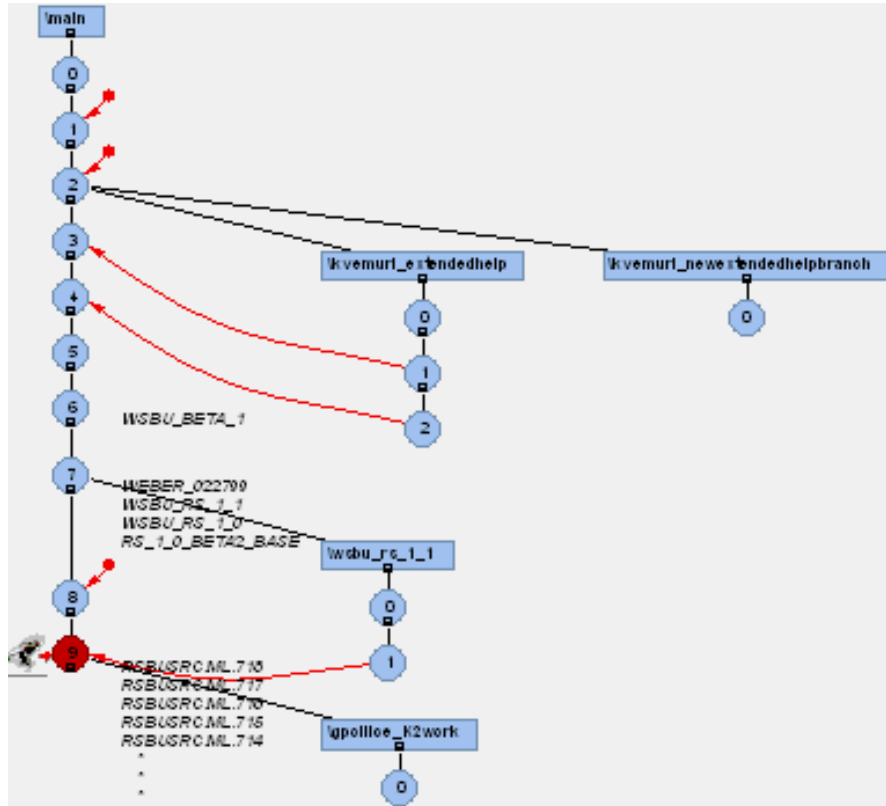
ClearCase stores information in a database, called a *VOB* (for versioned object base). A VOB contains *elements*; each element stores a project directory or file. For each element, ClearCase keeps track of each version you create. So, for example, in a fairly simple development environment, one element might have versions as shown in the *version tree* in Figure 1. (Each circle represents a version.)

Figure 1 Simple Version Tree



Another element might have a version tree as shown in Figure 2, representing the creation of more versions, development on separate branches (which start with rectangular boxes), and merges from one branch to another (denoted by an arrow from one version to another). The ability to develop on branches and then merge work together enables a powerful feature of ClearCase: the potential for parallel development.

Figure 2 More Complex Version Tree



In Figure 2 the *branch* structures demonstrate how ClearCase allows your team to work in parallel on the same element. Some groups use this branching feature to work on more than one release of software at a time. UCM uses branches to provide private work areas (development streams and views) and the public work area (the integration stream and view).

The arrows from one branch to another represent the *merge* actions, which allow you to reconcile differences between versions of the same element on different branches. These merges happen when you deliver work. ClearCase automates most merges, making it easy to share information developed on separate branches.

Views and Streams

So, if you're working with branches, how does ClearCase know which version of each element you want to work with? You use a ClearCase *view* to select one version of each element in your workspace. In UCM, your *stream* provides these configuration instructions to the view. (In base ClearCase, you must configure your view manually.)

When you join a project, UCM automatically configures your stream so that you see the right version for your work environment. A stream also tracks activities and baselines (the common starting place for members of your project).

Using ClearCase

Typically, you work with version-controlled files in a ClearCase view just as you work with other files on a networked disk drive. You can open them, print them, and so on. If you want to change a version-controlled file, however, you need to *check out* the file from the VOB. You can then edit the file and possibly unit test it. When you reach a milestone (even if you haven't completely finished working on the file), you *check in* the file. The chapter, *What Is UCM?* on page 13, describes how this check-out, change, check-in process fits into the general process of using UCM – working on activities and delivering work to the integration stream.

Behind the Scenes

Every UCM project must have a *PVOB* (a project VOB). A PVOB stores UCM objects, including one or more projects, and also streams, activities, and change sets. Typically, a project manager sets up a project's PVOB before ClearCase is installed on client computers. For more information, see *Configuring UCM: Planning and Setting Up Servers* on page 23.

ClearQuest Basics

ClearQuest helps you manage activities for your project. You can use ClearQuest to submit change requests, create queries to find change requests, and run reports or draw graphs describing change requests. Under UCM, an activity consists of a change request plus UCM-related information.

ClearQuest stores activities and change requests in a ClearQuest user database. An item in ClearQuest has a lifecycle, from submission through resolution. Each stage in the lifecycle is called a *state*.

A ClearQuest *schema* contains the information that defines how users work with items in ClearQuest. It includes type definitions, states, descriptions of forms to use, and so on. Typically, your project starts with a predefined schema included with ClearQuest. Your project manager may possibly modify the schema to reflect your project's needs. You store ClearQuest schemas in a *schema repository*.

For More Information

To learn more about the topics described in this chapter, see:

- *Introduction to ClearCase*, part of the ClearCase documentation set.
- *Introducing ClearQuest*, part of the ClearQuest documentation set.

Configuring UCM: Planning and Setting Up Servers

3

This chapter, and the chapters, *Configuring UCM: Setting Up the Project Manager's Computer* on page 29, and *Configuring UCM: Setting Up Client Computers* on page 49, describe how to configure a UCM environment using Rational ClearCase LT and Rational Suite on a local network.

Overview of Configuration

To set up a UCM environment using Rational Suite, you follow these high-level steps:

- 1 Plan your network environment.
- 2 Set up vendor databases for use with Rational tools.
- 3 Set up a ClearCase LT server.
- 4 Set up the project manager's computer.
- 5 Set up client computers.

The next few chapters describe each of these steps in more detail.

Planning a Network Environment

Before you can set up UCM, you need to plan how many servers and other computers you will use, and how you will make use of those computers.

Table 1 describes the network environment that we configure in this manual. We recommend that you decide how to lay out your network before you start setting up your UCM environment. When thinking about servers (and the *project-manager* computer), designate computers that are highly available and that are backed up regularly.

Table 1 Computers in a UCM Environment

Computer	Purpose	For More Information
<i>license-server</i>	One way to manage software licenses is to provide a license server.	Beyond the scope of this manual; for more information, see <i>Administering Licenses for Rational Suite</i> .
<i>database-server</i>	Server where vendor database software is installed; contains ClearQuest and RequisitePro databases.	See <i>Setting Up Vendor Databases</i> on page 25
<i>CCLT-server</i>	Server where you install ClearCase LT Server software; contains VOBs related to your UCM project.	See <i>Setting Up a ClearCase LT Server</i> on page 25
<i>project-manager</i>	Client computer containing the Rational project and ClearCase integration views; much UCM configuration occurs on this computer.	See <i>Configuring UCM: Setting Up the Project Manager's Computer</i> on page 29
<i>client</i>	Computer used by a developer; connects to UCM project and Rational project.	See <i>Configuring UCM: Setting Up Client Computers</i> on page 49

There are many correct ways to configure a UCM environment; these chapters describe one possible configuration using several computers. However, your environment may vary. You may be using more or fewer computers, but the details of the setup process remain as described in these chapters.

In this manual, we assume that you are using ClearCase LT, which requires you to work on a local network. On a geographically distributed network, you are likely to use ClearCase, which allows you to take advantage of MultiSite features. In that case, you can also take advantage of the MultiSite features offered by ClearQuest.

Setting Up Vendor Databases

With Rational RequisitePro and Rational ClearQuest, you have the option of using one of several types of vendor databases to store information. In this example, set up vendor databases on the *database-server* computer. To set up a vendor database, follow these general steps:

- 1 Select the type of vendor database to use.
- 2 Install the vendor database software.
- 3 Depending on the type of vendor database you selected, perform special configuration steps.

Table 2 describes where to find information about setting up vendor databases.

Table 2 Where to Find Information About Setting Up Vendor Databases

Product	See
Rational RequisitePro	“Configuring Enterprise Databases” in <i>Installing Rational RequisitePro</i>
Rational ClearQuest	“Configuring Vendor Databases” in <i>Installing Rational ClearQuest</i>

Setting Up a ClearCase LT Server

ClearCase LT is a networked product that allows for one server and multiple clients. The *server* is a host for VOBs (the repositories that contain project files and directories), storage directories, and other directories having to do with administering ClearCase. You can think of the server as the central area where project information resides. Developers use their *client* computers to get access to the project information and other files that reside on the server.

In this example, set up the ClearCase LT server on the *CCLT-server* computer. The remainder of this section gives an overview of setting up the ClearCase LT server.

Note: You can install and configure the ClearCase LT server once. If you have already performed these steps, you cannot perform them again.

Installing and Configuring the ClearCase LT Server

We recommend that you install and configure the ClearCase LT server before developers install ClearCase LT clients. To install the server, use the Rational Software setup program, as described in *Installing Rational Suite*. On the Choose Product page, select **Rational ClearCase LT Server** and configure a license.

If you do not configure a license, either at this point, or later, you will be unable to use Rational products.

Starting the ClearCase LT Server Setup Wizard

If you are logged in with appropriate permissions and group membership, the Server Setup Wizard starts as soon as the installation completes. If, instead, you see an error message, read it to learn how to log in so that you can continue the server setup. After you log in as the correct user, start the Server Setup Wizard: click **Start**, then click **Programs > Rational ClearCase LT Server > ClearCase Server Setup Wizard**.

In more detail, when you set up a UCM project, you become the owner of that project, and you are granted administrative privileges for that project. Also, to create a component VOB, you need to be a member of the same group that users of the VOB will be in. This restriction is explained more fully in the error message that appears if you are not logged in appropriately. For more details about group membership requirements, see *Installing Rational ClearCase LT*.

Working with the Server Setup Wizard

The Server Setup Wizard helps you set up the server's structure, but not the contents for your project. Specifically, the wizard helps you create:

- A project VOB (commonly called a *PVOB*)
ClearCase uses *PVOBs* to manage UCM projects. A *PVOB* contains UCM objects such as projects, activities, and change sets.
- An initial UCM project
A *UCM project* defines how developers get access to and update the items in a particular development effort.

- A component

You use a *component* to group a set of related artifacts (the files and directories) for use on a UCM project. When you use UCM with Rational Suite, a component is likely to contain code, requirements, tests, and visual models. The component includes an internal pointer to the PVOB. You can create more than one component for use with your project.

- A tutorial component

Work with the Server Setup Wizard as follows:

- 1 On the Create the ClearCase LT Storage Directory page, ClearCase recommends a storage area on an NTFS partition. You can change the location, but we recommend you use an NTFS partition to protect against accidental corruption.
- 2 On the Create ClearCase LT Source VOB page, enter a meaningful name that helps users understand the component VOB's contents. For example, you might want to name the component *GUI*, *admin-code*, or *libraries*.

On this page, select or clear the **Import Source Files** checkbox:

- If there are existing files that you'd like to place under ClearCase control, either files in another source control system or files not stored in any source control system, select the checkbox. If you select the box, ClearCase displays an import wizard.
 - If you do not want to import files or if this VOB will contain only requirements, test assets, or visual models, then clear the checkbox. The steps outlined in *Creating and Setting Up a Rational Project* on page 39 explain how to import files produced by Rational Suite tools.
- 3 On the Create Initial ClearCase LT Project page, provide a name for the UCM project. In this manual, we'll assume that you named your project *UCM-project*.
 - 4 On the "Will UNIX ClearCase LT Clients Use this Server" page, specify whether you have a Windows-only development environment or a mixed-platform environment.

You have successfully set up your ClearCase LT server.

For More Information

To learn more about the topics described in this chapter, see:

- *Installing Rational ClearCase LT*, in the ClearCase LT documentation set.
- *Installing Rational ClearQuest*, in the ClearQuest documentation set.
- *Installing Rational RequisitePro*, in the RequisitePro documentation set.
- *Installing Rational Suite*, in the Rational Suite documentation set.

Configuring UCM: Setting Up the Project Manager's Computer

4

Most of the work involved with configuring UCM concerns setting up the project manager's computer, which in this example we call *project-manager*. (For a list of computers involved with this example setup, see Table 1 on page 24.)

This chapter describes how to set up the *project-manager* computer by following these high-level steps:

- 1 Install Rational Suite.
- 2 Configure ClearQuest.
- 3 Configure ClearCase LT.
- 4 Create a Rational project.
- 5 Optionally, configure the RequisitePro / ClearQuest integration.

Installing Rational Suite

Install an edition of Rational Suite on the *project-manager* computer (the computers used in this setup are described in Table 1 on page 24. Use the Rational Software setup program, as described in *Installing Rational Suite*. On the Choose Product page, select the edition of Rational Suite you purchased and configure a license.

Configuring ClearQuest

You can use UCM with or without ClearQuest. Without ClearQuest, ClearCase manages activities, creating them at checkout time. Each ClearCase activity tracks a *change set* – the set of files affected by a single change to the system – and has a name.

ClearQuest adds more process support for activities. For example, when you use ClearQuest with UCM, you can assign and schedule activities and you can monitor project status. ClearQuest maintains information about an activity's state, priority, severity, assigned user, and so on.

This manual assumes that you have chosen to use UCM with ClearQuest. Setting up ClearQuest involves these steps:

- 1 Set up a schema repository.
- 2 Set up a user database. When you configure the ClearCase LT server, you will be prompted to provide the name of this user database. (The UCM integration links your component to the ClearQuest user database.)

Setting Up a Schema Repository

ClearQuest stores its schemas – the metadata that defines how users work – in a *schema repository*. After you create a schema repository, you can create one or more *user databases*, which store project activities.

After you associate a user database with a schema, you can upgrade it only with a newer version of the same schema. You can't change the schema and you can't use an older version of the schema.

Set up the schema repository as follows:

- 1 Start the ClearQuest Maintenance tool: click **Start**, then choose **Programs > Rational Suite > Rational ClearQuest > Rational ClearQuest Maintenance Tool**.
- 2 On the first page, choose **Create a new schema repository**.
- 3 On the Create a New Schema Repository page, choose the type of vendor database and supply information related to that database type. The “Installing ClearQuest for an Administrator” chapter in *Installing Rational ClearQuest* contains details about how to enter the information on this page. (The information changes depending on the type of vendor database you are using.)
- 4 On the Create Sample Database page, you have the option of creating a sample user database. You may want to create this database if you'd like to experiment with ClearQuest before using it for production purposes. If you do create the sample database, choose the schema to use with the database, following the guidelines in *Setting Up a User Database* on page 31.
- 5 If you choose to create a sample database, enter information about the vendor database on the Create a Sample User Database page.
- 6 On the Save Current Setting for Later Use page, you can optionally choose to save the information you entered in a profile file. If you choose to save the profile file, users can later connect to the sample database simply by entering a path to the profile file. Otherwise, users can configure their user database manually.

Setting Up a User Database

The user database contains activity records (and other types of records) and uses one of the schemas that resides in the schema repository.

Guidelines for Choosing a Schema

Pick the schema you want to use with UCM. Two schemas, Enterprise and UnifiedChangeManagement, are configured to work with UCM. Alternately, you can load UCM packages into any other schema, as described in *Managing Software Projects with ClearCase*.

Note: A ClearCase LT server can use exactly one schema. The ClearQuest schema repository can contain multiple schemas. Associating the first ClearQuest user database with ClearCase LT (described on page 33) determines which schema will be used with the ClearCase LT server.

Creating a User Database

To create a user database, you use the ClearQuest Designer, as follows:

- 1 Start the ClearQuest Designer: click **Start**, then click **Programs > Rational Suite > Rational ClearQuest > Rational ClearQuest Designer**.
- 2 On the ClearQuest Designer Login dialog box, click **OK** to log in using the default user name, **admin**, and the default password, which is blank.
- 3 On the Open Schema wizard, click **Cancel**.
- 4 From the ClearQuest Designer menu, choose **Database > New Database**.
- 5 On the New Database wizard:
 - a Supply a logical database name with five or fewer characters.
 - b Enter properties of the physical database.
 - c Declare how long a user can maintain an idle connection to the database.
 - d Choose a schema with which to associate the database.
- 6 Leave the ClearQuest Designer open.

Adding Users to the User Database

Add users to the user database, as described in “Administering Users,” in *Administering Rational ClearQuest*.

Optional: Configuring the User Database

You can choose to perform the tasks in this section now or after you configure the ClearCase LT server.

When you set UCM policies (on page 33), you have the option of setting policies specific to ClearQuest as well. If you choose the policy **Do ClearCase Action After Delivery**, then by default, when developers deliver an activity, its state changes to Complete.

If the default action requires entries in certain fields of the activity's record and one of those fields is empty, the script returns an error and leaves the deliver operation in a noncompleted state. To recover from the error, the developer needs to fill in the required fields in the activity's record and resume the deliver operation.

To work around this possible problem, you can take one of the following actions:

- Change the field so that it is Optional (read about “Customizing a Schema” in *Administering Rational ClearQuest*.)
- Write hook code that completes the required field (read about “Using Hooks to Customize Your Workflow” in *Administering Rational ClearQuest*.)

Configuring ClearCase LT

In this section, you join the UCM project you created in *Configuring UCM: Planning and Setting Up Servers* on page 23, and configure ClearCase LT as follows:

- Set UCM project policies
- Configure streams and create views
- Share the view directories so that the views are available across the network
- Optionally add components to the project

Setting UCM Policies

As part of setting up the ClearCase LT server, you provided a name for the UCM project (*UCM-project*; you performed this work on page 27). Use the *UCM-project* Properties dialog box to set project policies. To display this dialog box:

- 1 Start the ClearCase Explorer: click **Start**, then click **Programs > Rational Suite > Rational ClearCase LT > Rational ClearCase LT Explorer**.
- 2 From the ClearCase Explorer, on the **Toolbox** tab, in the **UCM** area, click **Project Explorer**.
- 3 On the ClearCase Project Explorer, in the left pane, right-click a project, and from the shortcut menu, choose **Properties**.

To set UCM project policies:

- 1 On the *UCM-project* Properties dialog box, on the **Policy** tab:
 - See *Promoting the Baseline and Working with Promotion Levels* on page 67 for information about recommended baselines.
 - A user must always check in all files related to an activity before delivering that activity. If you enable the policy, **Allow deliveries from streams with pending checkouts**, then a user can deliver work in one activity even if there are checkouts related to other activities. If you disable the policy, then, before delivering work, the user must check in all files in that stream.
 - If you enable the policy, **Require development streams to be rebased to the current recommended baselines before delivery**, then much of the integration work involved with delivery (merging and testing) will occur in the developers stream, rather than in the integration stream. If you disable the policy, it is more likely that this work will occur in the integration stream.
 - In the “Make the following components modifiable” area, you can declare that component VOBs associated with this project are read-write or read-only.
 - To learn more about each of these policies, click **Help** on the **Policy** tab.
- 2 On the *UCM-project* Properties dialog box, on the **ClearQuest** tab:
 - a Select **Project is ClearQuest-enabled**.
 - b In the Link to this ClearQuest User Database, select the name of the database you created in *Creating a User Database* on page 31.
 - c At the ClearQuest Login prompt, log in to the ClearQuest user database, using a valid user name.

- d Click **Help** to learn about ClearQuest policies, and decide whether to enable any of them.

Note: If you enable the **Do ClearQuest Action After Delivery** policy, read about the additional tasks you may need to perform in *Optional: Configuring the User Database* on page 32.

- 3 Click **OK** on the *UCM-project* Properties dialog box.

Joining the Project

From the *project-manager* computer, you need to declare that you will be using the *UCM-project*, by *joining the project*:

- 1 Start the ClearCase Explorer: click **Start**, then click **Programs > Rational Suite > Rational ClearCase LT > Rational ClearCase LT Explorer**.
- 2 In the leftmost pane, on the **Toolbox** tab, click the **UCM** button, then click **Join Project**. The Join Project wizard appears.
- 3 On the Choose a Project page, select *UCM-project*.
- 4 On the Join Project wizard, click **Next** to display the Create a Development Stream page.

Configuring Streams and Creating Views

The development stream, combined with a development view, provide an area for developers to work on activities in isolation from each other.

- 1 On the Create a Development Stream page, ClearCase chooses the name of your development stream. You can change the name of this stream if you want.

ClearCase tells you the name of the integration stream: there is only one integration stream per project. The integration stream was created when the ClearCase LT server was configured on page 25.

- 2 On the Choose Location page, ClearCase supplies a location on your system for the development view. You can change this location and optionally enter a comment describing the view.
- 3 On the next Choose Location page, ClearCase supplies a location on your system for an integration view associated with the project's integration stream. Again, you can change the location and optionally enter a comment.

- 4 On the Choose Components page, you can select the components to copy to your development view.

Remember that your development view is a private work area and the integration view reflects the contents of the shared work area. Therefore, you will have two versions of each file on your system. Initially, the two versions will be identical, but they will diverge as you make changes in your development view and other developers deliver changes to the integration stream.

- 5 The Confirm page reminds you that you are about to create two views and allows you to see details about each.

ClearCase creates the views, loads the components you specified into those views, and adds a button named after your project to the **Views** tab of the ClearCase Explorer. In this new project area, it places shortcuts to the two views you created.

Sharing the Integration View Directories

For each integration view associated with your project, you must create a share name for the view directory so that other developers can get access to project assets from their computers.

To create a share name, perform these steps for each integration view associated with your project:

- 1 Use Windows Explorer to navigate to the view directory. *Hints:* This is the directory above the component you created. Also, in the View Creation Wizard, on the Choose Location page, you chose this directory in answer to the question “Where would you like the root of this view?”
- 2 Right-click the directory, and from the shortcut menu, choose **Sharing**.
- 3 On the directory’s Properties page, on the **Sharing** tab:
 - a Select **Shared As**.
 - b In the **Share Name** field, enter a meaningful name describing the view, for example, *project_integration_main* or *project_integration_requirements*.
 - c Click **OK**.

Now, users and other programs can get access to these project views by using a UNC pathname, for example, `\\project-manager\project_integration_main`.

Creating Additional Integration Views

We recommend that you create additional integration views to avoid locking problems during checkin and deliver operations. When you use UCM to store RequisitePro and Test assets, the project manager checks out all assets, developers work on the assets, and the project leader performs a Check In All operation. During Check In All, ClearCase checks in all the assets and checks them out again. While the Check In All operation is proceeding, the ClearCase view is unavailable for other work in that component.

To work around this potential problem, follow this plan:

- Use the integration view you created on page 34 for work with Rational Rose, code, and other assets developed outside of Rational Suite.
- Create two additional integration views, one each for Rational RequisitePro and Rational Test. Remember to share these views, as described on page 35.

For each additional integration view that you want to create:

- 1 Start the ClearCase Project Explorer: click **Start**, then click **Programs > Rational Suite > Rational ClearCase LT > Project Explorer**.
- 2 In the left pane, expand the top-level projects folder, and select your project.
- 3 In the right pane, right-click the Integrations stream. From the shortcut menu, select **Create View**.
- 4 In the View Creation wizard:
 - a On the first page, select **Create an integration view**.
 - b On the Choose Location page, choose a name that describes the purpose of the view, for example:
disk:\view-directory\project_integration_requirements or
disk:\view-directory\project_integration_tests
 - c On the Choose Components page, select the components to copy to your view.
 - d The Confirm page reminds you that you are about to create an integration view and allows you to see details about the view.
- 5 Share the view directories, as described on page 35.

Adding Another Component to the Project

You may want to associate more than one component with a UCM project. For example, you might want to store requirements, test assets, and visual models in one component, *rtvm*. You might then decompose your project into subsystems, say, GUI, admin, libraries, and so on. Each subsystem maps to a component, say *user-interface*, *admin-code* and *libraries*. Each component would store artifacts, such as code and documentation, related to the subsystem.

In this section, assume that you've already created one component, *rtvm*. You are now ready to add another component, *admin-code*. Note that you can add components either when you're first establishing a project, or later, when the scope of your project broadens.

Creating the Component

Assume that during server setup, you created the *rtvm* component. To add the second component, *admin-code*, to your project:

- 1 Click **Start**, then click **Programs > Rational Suite > Rational ClearCase LT > Create VOB**. The VOB Creation wizard appears.
- 2 On the VOB Creation wizard:
 - a Provide a name for the new component, for example, **admin-code**.
 - b Optionally, provide a comment.
- 3 Read the confirmation information.

ClearCase creates the component and makes it available to the project, but you need to perform some further configuration before you can use it:

- Add the baseline to the integration stream
- Update development and integration views

Adding the Component's Baseline to the Integration Stream

Configure the integration stream so it includes the new component's baseline.

- 1 Start the ClearCase Project Explorer: click **Start**, then click **Programs > Rational Suite > Rational ClearCase LT > Project Explorer**.
- 2 In the Project Explorer's tree browser, open the **projects** folder, then select your project. An **Integration stream** icon appears in the right pane.
- 3 Right-click the **Integration stream** icon and from the shortcut menu, select **Properties**. The Integration Stream Properties sheet appears.

- 4 On the Properties sheet, on the **Configuration** tab, click **Add** below Foundation baselines. You will add a baseline associated with the new component, effectively making the new component available to the project.
- 5 In the Add Baseline dialog box:
 - a Select the new component name from the **Component** list.
 - b In the Baselines area, select the baseline associated with the new component.
 - c Click **OK**.The Integration Stream Properties sheet now lists both baselines from the two components.
- 6 Click **OK** to close the Integration Stream Properties sheet, then click **OK** to close the Confirm message box.
- 7 ClearCase offers to rebase your stream. Click **OK**.
- 8 When the rebase finishes, click **Complete**, then click **Close**.

Updating Development and Integration Views

This section applies if you are using ClearCase LT or if you are using ClearCase with snapshot views. For each integration or development view associated with the project, edit the view's load rules to include the new component. (Notify developers to complete this step, also.) This action instructs your view to make the new component available to you.

To edit the load rules for a view:

- 1 Start the ClearCase Explorer: click **Start**, then click **Programs > Rational Suite > Rational ClearCase LT > ClearCase LT Explorer**.
- 2 In the left pane, on the **Views** tab, if your project is not listed, click **View > Refresh View Shortcuts**.
- 3 Click your project's button to display the views associated with your project.
- 4 Right-click a view and from the shortcut menu, click **View Properties**.
- 5 On the **Load Rules** tab, click **Edit load rules**. The Choose Elements to Load dialog box appears.

- 6 On the Choose Elements to Load dialog box:
 - a If you don't see the new component in the Available Elements list, select **Show All VOBs**.
 - b From the Available Elements list, select the component you are adding and click **Add**. The component appears in the Selected Elements list.
 - c Click **OK**.
- 7 Click **OK** to quit the Properties sheet. ClearCase displays the Start Update dialog box.
- 8 Click **OK** to start the update, and when the update completes, review the results.

Creating and Setting Up a Rational Project

A Rational project associates Rational Suite products so that you can take advantage of integrations among those products. To create and set up a Rational project, you use the Rational Administrator. In addition, you use the Rational Administrator to enable a Rational project for use with UCM.

Creating a Rational Project

This section describes how to create a Rational project. If you used an earlier version of Rational Suite, you may have an existing repository that you need to upgrade. (A repository is now called a project.) For more information about upgrading a repository, see *Using the Rational Administrator* or read about upgrading in *Installing Rational Suite*.

Creating a Share Name

Before you create the Rational project, create a share name pointing to the directory that will contain the project's files:

- 1 Create a directory on the *project-manager* computer, where you will create the Rational project. Choose an area that is not under version control. In other words, the area should not be part of a VOB.
- 2 Create a share name pointing to the directory you created in Step 1:
 - a In Windows Explorer, navigate to the directory.
 - b Right-click the directory, and from the shortcut menu, choose **Sharing**.

- 3 On the directory's Properties page, on the **Sharing** tab:
 - a Select **Shared As**.
 - b In the **Share Name** field, enter a meaningful name describing the directory, for example, *rational-project*.
 - c Click **OK**.

Now, users and other programs can get access to these project views by using a UNC pathname, for example, `\\project-manage\rational-project`.

Creating the Rational Project

To create the Rational project:

- 1 Start the Rational Administrator: Click **Start**, then click **Programs > Rational Suite > Rational Administrator**.
- 2 Start the New Project Wizard by clicking **File > New Project** on the Administrator menu.
- 3 On the wizard's first page:
 - a Supply a name for your project, for example, *revenue*. The dialog box prevents you from typing illegal characters.
 - b In the Project Location field, specify a UNC path to the root of the project, referring to the share name you created in *Creating a Share Name* on page 39. For example, `\\project-manage\rational-project\revenue`. Make sure that you specify a path that is at least in a subdirectory of the share; in this example, *rational-project* is the share, so you need to specify a directory at least one level lower.
 - c Select **Use ClearCase and Unified Change Management to baseline project assets**.
- 4 On the Project UCM Policies page:
 - a Determine whether you want requirement assets and test assets under version control. If so, select the appropriate checkboxes.
 - b Then, from the tree under Associated UCM Project, expand the **projects** folder and select the UCM project with which to associate this Rational project.
- 5 On the Security page, decide whether you want to associate a password with the Rational project. If so, supply the password. Otherwise, leave the fields blank on this page.
- 6 On the Summary page, select **Configure Project Now**.

The Configure Project dialog box appears.

Configuring a Rational Project

When you configure a Rational project, you associate the project with items of the following types:

- One RequisitePro project
- One test datastore
- One ClearQuest database
- One or more Rose models

If the Configure Project dialog box is not already started, display it as follows:

- 1 In the Rational Administrator, select the project you want to configure and click **File > Connect**. Optionally supply a password.
- 2 Select the project to configure and click **File > Configure Project**.
- 3 If you see an error from ClearQuest, then start the Task Manager and end the CQintsvr11.exe task.

Associating Requirements

A *RequisitePro* project organizes and helps you manage a related set of requirements. Each RequisitePro project includes a requirements database, documents, discussions, histories, and other management-related and administrative items.

Go to the section most appropriate to your situation, as described in the following table.

If	Then follow the instructions in
There is no RequisitePro project you would like to associate with your Rational project	<i>Creating a New RequisitePro Project</i> on page 42
You have already created a UCM-enabled Rational project and associated with it a RequisitePro project that you did not place under version control	<i>Placing a Previously Associated RequisitePro Project Under Configuration Management</i> on page 42
There is an existing RequisitePro project you want to associate with your Rational project	<i>Associating an Existing RequisitePro Project with the Rational Project</i> on page 43

Creating a New RequisitePro Project

To create a new RequisitePro project:

- 1 On the Configure Project dialog box, click **Create** in the Requirements Assets area. RequisitePro displays the Create Rational RequisitePro Project dialog box.
- 2 Select a Project Template, for example, *Use-Case Template*, then click **Create**.
- 3 On the Project Properties dialog box:
 - a In the Name field, specify a name for the RequisitePro project.
 - b In the Directory field, use a UNC pathname to specify an area in the integration view where you'd like the requirements to reside. For example, you might specify `\\project-manage\project_integration_requirements\requirements`.
 - c In the Database field, specify the type of database you plan to use with your RequisitePro project. You configured this database in *Setting Up Vendor Databases* on page 25.
 - d Click **Properties** (next to the Database field) to provide a database configuration. Use online Help to discover how to configure the database.
 - e Optionally, provide a comment describing the RequisitePro project.
 - f Click **OK**. RequisitePro creates the project and adds it to version control.

Placing a Previously Associated RequisitePro Project Under Configuration Management

In some cases, you may have already created a UCM-enabled Rational project and associated with it a RequisitePro project that you did not place under version control. In this case, in the Requirements Assets area of the Configure Project dialog box, an **Add to CM** button appears.

Click **Add to CM** to place the RequisitePro project under version control.

Associating an Existing RequisitePro Project with the Rational Project

To associate an existing RequisitePro project with your project:

- 1 On the Configure Project dialog box, click **Select** in the Requirements Assets area. The Rational RequisitePro Project Relocation dialog box appears.
- 2 Take one of the following steps depending on the message displayed in the Project Relocation dialog box:
 - If the RequisitePro project is already located in a valid integration view, then you can leave it there or choose another location in the view where you'd like to relocate the project.
 - If the RequisitePro project is not currently under version control, then choose a location in the integration view where you'd like to relocate the project.

In either case, you might want to relocate the RequisitePro project to a new directory named \requirements. When you click **OK**, RequisitePro relocates the project to the location you specified.

Associating Test Assets

Go to the section most appropriate to your situation, as described in the following table.

If	Then follow the instructions in
There is no Test datastore you would like to associate with your Rational project	<i>Creating a New Test Datastore on page 44</i>
You have already created a UCM-enabled Rational project and associated with it a test datastore that you did not place under version control	<i>Placing a Previously Associated Test Datastore Under Configuration Management on page 44</i>
There is an existing Test datastore you want to associate with your Rational project	<i>Associating an Existing Test Datastore with the Project on page 44</i>

Creating a New Test Datastore

To create a new test datastore:

- 1 On the Configure Project dialog box, click **Create** in the Test Assets area. The Create Test Datastore tool appears.
- 2 On the Create Test Datastore dialog box:
 - a In the New Test Datastore Path field, use a UNC pathname to specify an area in the integration view where you'd like the tests to reside. For example, you might specify `\\project-manager\project_integration_tests\tests`.
 - b Select initialization options as appropriate.
 - c Click **OK**. Rational Test creates a test datastore and adds it to version control.

Placing a Previously Associated Test Datastore Under Configuration Management

In some cases, you may have already created a Rational project and associated with it a test datastore that you did not place under version control. In this case, in the Test Assets area of the Configure Project dialog box, an **Add to CM** button appears.

Click **Add to CM** to place the test datastore under version control.

Associating an Existing Test Datastore with the Project

To associate an existing test datastore with your project:

- 1 Move the test datastore to an area in your integration view. For example, under the VOB directory, you might create a subdirectory named `\tests` and place the test datastore there.
- 2 On the Configure Project dialog box, click **Select** in the Test Assets area, then navigate to the directory you worked with in Step 1.

Associating Change Requests

If you set up ClearQuest as described in this chapter (that is, you created a ClearQuest user database, then associated it with your UCM project), ClearQuest is already associated with your Rational project. You may have noticed a message to this effect when you created the Rational project. The message says that the ClearQuest database would be automatically associated with the Rational project. In this case, the Configure Project dialog box should contain the name of the ClearQuest database in the Change Requests area.

If you perform the setup steps in a different order and the ClearQuest user database is not already associated with the Rational project, associate the ClearQuest user database with the Rational project as follows:

- 1 On the Configure Project dialog box, click **Select** in the Change Requests area.
- 2 Browse to the ClearQuest user database you created, as described in *Configuring ClearQuest* on page 29.

Associating Rose Models

Before you associate Rose models with the project, read *Managing Rose Model Files*, which describes at a high level how to break the model into smaller units before placing it under version control.

Managing Rose Model Files

By default, Rose saves a complete model as a single model (.mdl) file. However, when many users work on a model at the same time, you can reduce contention and enable parallel development by breaking the model into a series of individual files called *controlled units*.

Controlled units are the configuration elements that a team places under version control. When using controlled units, each team or team member is responsible for maintaining or updating a specific unit.

For more information and guidelines about working with controlled units, read about breaking a model into controlled units in *Guide To Team Development, Rational Rose*.

Establishing a Virtual Path Map in Rose

Rose maintains internal absolute pathnames to models and parts of models. This means that in order to share a Rose model with another developer, you need to use exactly the same paths, including drive letters, to get access to the same Rose model. To work around this restriction, set up a virtual path map as follows:

- 1 Start Rose: click **Start**, then click **Programs > Rational Suite > Rational Rose**.
- 2 Open the Rose model you will be working with.
- 3 From the Rose menu, choose **File > Edit Path Map** to display the Virtual Path Map dialog box.

- 4 On the Virtual Path Map dialog box:
 - a In the Symbol field, type **CURDIR**.
 - b In the Actual Path field, type **&**.
 - c Click **Add**.

The following line appears in the *Virtual Symbol to Actual Path Mapping* list:

```
$CURDIR      =&
```

- d Click **Close**.
- 5 Quit Rose.
- 6 Start Rose again, open the Rose model, and quit Rose.

For more information about virtual path maps, view the online Help for the Virtual Path Map dialog box.

Associating Rose Files with the Rational Project

To associate an existing Rose model with the Rational project:

- 1 Move the Rose files to the integration view for your project. For example, under the VOB directory, you might create a subdirectory named `\models` and place the visual models there.

Hint: You can move the files using the Windows Explorer, or you can use Rose to open the model, then use **File > Save As** to save the model files to a new location in your integration view.

- 2 In Rose:
 - a Open the Rose model, which is now in the integration view.
 - b In the Rose browser or in a model diagram, select one or more packages and right-click. From the shortcut menu, select **Add to Version Control**.
 - c On the Select Files To Be Controlled page, select specific files or click **Select All**, then click **OK**.
 - d When the Add to Source Control dialog box appears,
 - Accept the default activity.
 - Clear the **Keep checked Out** check box.
 - Click **Apply to All**.
 - e You can quit Rose now if you do not need to continue using it.

- 3 On the Configure Project dialog box, click **Add** next to Models.
- 4 Browse to the model you just added to version control.

Creating a Baseline

After you set up the Rational project and place items under version control, create a baseline to provide a common starting point for your users. To create a baseline, follow these high-level steps in the ClearCase Project Explorer:

- 1 Lock the Integration Stream.
- 2 Create the Baseline.
- 3 Unlock the Integration Stream.
- 4 Promote the Baseline.

For more detailed instructions about creating and working with baselines, see *Working with Baselines* on page 63.

Updating Integration Views

Update the integration views on the *project-manager* computer:

- 1 Start the ClearCase Explorer. On the **Views** tab, under the button for your project, determine whether the integration views are listed. If not, from the menu, select **View > Refresh**.
- 2 For each integration view associated with the UCM project, right-click a view. From the shortcut menu, select **Update View**.

Configuring the RequisitePro / ClearQuest Integration

If you want to enable the integration between ClearQuest and RequisitePro, read about “Configuring the Integration for ClearQuest and RequisitePro” in *Configuring Rational Suite*.

For More Information

To learn more about the topics described in this chapter, see:

- *Installing Rational Suite, Using the Rational Administrator, and Configuring Rational Suite* in the Rational Suite documentation set.
- *Installing Rational ClearQuest and Administering Rational ClearQuest* in the ClearQuest documentation set.
- *Managing Software Projects with ClearCase* in the ClearCase or ClearCase LT documentation set.
- *Guide To Team Development, Rational Rose*, in the Rose documentation set.

Configuring UCM: Setting Up Client Computers

5

This chapter describes how to set up a computer for a typical developer – someone who creates code, or who uses Rational Suite to develop requirements, visual models, or test assets. To set up a client environment, follow these steps:

- 1 Install Rational Suite.
- 2 Register the Rational project.
- 3 Join the UCM project.
- 4 Establish a ClearQuest connection.

Installing Rational Suite

Install Rational Suite on your system after:

- Your system manager has configured a Rational Suite environment.
- Your project manager has ensured that the tasks described in *Configuring UCM: Planning and Setting Up Servers* on page 23 and *Configuring UCM: Setting Up the Project Manager's Computer* on page 29 have been performed.

Run the Rational setup program, as instructed by your system manager or project leader:

- You will be prompted to enter the name of the ClearCase LT Server. Specify the server used by your team. (In this manual, it is *CCLT-server*.)
- Ensure you have a valid license for your Suite product.

Registering the Rational Project

Register the Rational project used by your team, as follows:

- 1 Start the Rational Administrator: click **Start**, then click **Programs > Rational Suite > Rational Administrator**.
- 2 From the Rational Administrator menu, choose **File > Register Project**.

- 3 In the Select Rational Project dialog box, in the File Name field, enter a UNC pathname to the project file, for example,
`\\project-manager\rational-project\revenue\revenue.rsp`
Hint: Once you enter the beginning of the UNC pathname, for example, `\\project-manager`, you can click **Open** and browse to complete the pathname.
- 4 In the left pane of the Rational Administrator, select the project you just registered. In the right pane, verify that the location and path are expressed as UNC pathnames.
- 5 Close the Rational Administrator.

Joining the UCM Project

Now use the ClearCase Explorer to join the UCM project. As part of this activity, you are prompted to set up your ClearCase environment by configuring streams and views.

- 1 Start the ClearCase Explorer: click **Start**, then click **Programs > Rational Suite > Rational ClearCase > ClearCase Explorer**.
- 2 In the leftmost pane, on the **Toolbox** tab, click the **UCM** button, then click **Join Project**. The Join Project wizard appears.
- 3 On the Choose a Project page, expand the project folder and select the correct project for your team.
- 4 On the Create a Development Stream page, ClearCase chooses the name of your development stream. You can change the name of this stream if you want.

The development stream, combined with a development view, provide an area for you to work on activities in isolation from your team.

ClearCase tells you the name of the integration stream: there is only one integration stream per project. The integration stream was created when the ClearCase LT server was configured on page 25.

- 5 On the Review Types of Views page, read the description of the development view and the integration view.
- 6 On the Choose Location page, ClearCase supplies a location on your system for the development view. You can change this location and optionally enter a comment describing the view.

- 7 On the next Choose Location page, ClearCase supplies a location on your system for an integration view associated with the project's integration stream. Again, you can change the location and optionally enter a comment.
- 8 On the Choose Components page, you can select the components to copy to your development view.

Your development view is a private work area and the integration view reflects the contents of the shared work area. Therefore, you will have two versions of each file on your system. Initially, the two versions are identical, but they diverge as you make changes in your development view and other developers deliver changes to the integration stream.

- 9 The Confirm page reminds you that you are about to create two views and allows you to see details about each.

ClearCase creates the views, loads the components you specified into those views, and adds a button named after your project to the **Views** tab of the ClearCase Explorer. In this new project area, it places shortcuts to the two views you created.

Registering a Rational Project

When you register a Rational project on your computer, you make that project available for your use. To register the Rational project:

- 1 Ask your project manager for the location of the Rational project. The location is a UNC pathname, which has the format `\\project-manager\rational-project\subdirectory\project.rsp`. Your project manager established the location when he or she created the project (see *Creating the Rational Project* on page 40).
- 2 Start the Rational Administrator: click **Start**, then click **Programs > Rational Suite > Rational Administrator**.
- 3 From the Rational Administrator menu, choose **File > Register Project**.
- 4 In the Select Rational Project dialog box, browse to the Rational project you plan to use, remembering to use a UNC pathname, then click **Open**.

The project name appears in the Rational Administrator's left pane.

Establishing a Virtual Path Map in Rose

Perform the steps in this section before you use Rational Rose on your computer. Rose maintains internal absolute pathnames to models and parts of models. This means that in order to share a Rose model with another developer, you need to use exactly the same paths, including drive letters, to get access to the same Rose model. To work around this restriction, set up a virtual path map as follows:

- 1 Start Rose: click **Start**, then click **Programs > Rational Suite > Rational Rose**.
- 2 From the Rose menu, choose **File > Edit Path Map** to display the Virtual Path Map dialog box.
- 3 On the Virtual Path Map dialog box:
 - a In the Symbol field, type **CURDIR**.
 - b In the Actual Path field, type **&**.
 - c Click **Add**.

The following line appears in the *Virtual Symbol to Actual Path Mapping* list:

```
$CURDIR      =&
```
 - d Click **Close**.
- 4 Quit Rose, then restart it before you perform any work in Rose.

For more information, view the online Help for the Virtual Path Map dialog box.

Establishing a ClearQuest Connection

Finally, connect to the ClearQuest database used by your team:

- 1 Start ClearQuest: click **Start**, then click **Programs > Rational Suite > Rational ClearQuest**.

ClearQuest displays the Connect to an Existing Schema Repository dialog box.
- 2 If your project leader has given you a path to a ClearQuest profile file:
 - Then click **Load Profile** and browse to the location of the profile file.
 - Otherwise, enter information about the Schema Repository you are using.
- 3 Click **Finish**.
- 4 When you are prompted to connect to the repository, click **Done**.

You have now set up your workstation, and you are ready to work on your project.

This chapter describes the process of developing software while using UCM. The process differs depending on whether you are:

- Developing requirements or test assets
- Developing visual models or code

The remainder of this chapter describes these two styles of working.

Note that you can store any type of artifact or file with UCM and ClearCase, but that broader topic is beyond the scope of this manual. For more information about working with other types of files (documentation, specifications, and so on), see *Managing Software Projects with ClearCase*.

Developing Requirements and Test Assets

In the current version of Rational Suite, the members of your team who develop requirements and test assets work outside of ClearCase. In other words, these team members develop the project's requirements and test assets without going through ClearCase – they accomplish their work as they would normally in a non-version-controlled environment.

At certain points in the development cycle, as dictated by your development process, your project manager checks in all requirements or test assets, and then creates a new baseline. For more information about these tasks, see *Working with Baselines* on page 63.

Developing Rose Visual Models and Code

You can use the UCM model fully when you work with Rose visual models or with code. (See *What Is UCM?* on page 13 for an overview of the UCM model.) This section describes the following work:

- Joining a project
- Working on an activity
- Delivering an activity
- Rebasng your development stream

Starting the ClearCase Explorer

To perform the work described in this section, you typically use the Rational ClearCase Explorer or a menu item in your development environment, either Rational Rose, or a code development environment (an IDE) from another company. If you use an IDE, and if it conforms to the Microsoft Source Code Control (SCC) API, then you can use ClearCase commands directly from the IDE.

To start the ClearCase Explorer from your desktop, click **Start**, then click **Programs > Rational Suite > Rational ClearCase LT > ClearCase Explorer**.

Joining a Project

To contribute work to a UCM project, you first set up two work areas: a development area for working on activities in isolation from your team and an integration area for testing the activities you deliver. You perform this task (called *joining a project*) once, the first time you start work on a project.

To join a project:

- 1 In the ClearCase Explorer, in the left pane, on the **Toolbox** tab, click the **UCM** button, then click **Join Project**. The Join Project wizard appears.
- 2 On the Choose a Project page, expand the top-level folder (by default, **projects**), and select the project you want to join.
- 3 On the Create a Development Stream page, ClearCase chooses the name of your development stream (which is part of your private work area). You can change the name of this stream if you want.

ClearCase tells you the name of the integration stream: there is only one integration stream per project. The integration stream was created when your project manager set up your project.

- 4 On the Review Types of Views page, read the description of the development view and the integration view.
- 5 On the Choose Location page, ClearCase supplies a location on your system for the development view. You can change this location and optionally enter a comment describing the view.
- 6 On the next Choose Location page, ClearCase supplies a location on your system for an integration view associated with the project's integration stream. Again, you can change the location and optionally enter a comment.

- 7 On the Choose Components page, you can select the components to copy to the views you are creating. Remember that your development view is a private work area and the integration view reflects the contents of the shared work area. Therefore, you will have two versions of each file on your system. Initially, the two versions are identical, but they diverge as you make changes in your development view and other developers deliver changes to the integration stream.
- 8 The Confirm page reminds you that you are about to create two views and allows you to see details about each.

ClearCase creates the views, loads the components you specified into those views, and adds a button named after your project to the **Views** tab of the ClearCase Explorer. In this new project area, it places shortcuts to the two views you created.

Working On an Activity

To work on an activity, you:

- Set the activity to work on.
- Check out the file or files that make up the activity's change set.
- Make changes.
- Check in the file or files.

Setting an Activity: With ClearQuest

If your project uses ClearQuest with UCM, an activity is created when someone creates a new record (such as a defect or enhancement request), associates it with a UCM project, and assigns it to an owner.

Use the ClearCase Explorer to identify activities assigned to you and to select the activity to work on:

- 1 In ClearCase Explorer, on the **Views** tab, in your project area, click your development view. The tree browser and details panes display the hierarchy for your view.
- 2 In the tree browser, beneath the view name, click **My Activities**. The right pane lists the activities assigned to you.

- 3 Click the checkbox next to the activity you want to work on. This action *sets* the current activity to this activity.

You must set a current activity in your view before you can check out and modify files. Setting a current activity instructs ClearCase to assign any versions of files you create to that activity's change set.

To view more information about an activity, right-click it and choose **Properties** from the shortcut menu.

Setting an Activity: Without ClearQuest

If you are not using ClearQuest with UCM, you set the activity at checkout time. From the Check Out dialog box, you can select or create the activity to work with.

Checking Out Files

To check out a file:

- In the ClearCase Explorer, go to your development view, navigate to the file you want to change, and from the menu, click **Tools > Checkout**.
- From the Rose menu, click **Tools > ClearCase > Check Out**, then select the files to check out.

If you are not working with ClearQuest, select or create an activity to work on.

Guidelines for Working with Rose Files

In the section, *Managing Rose Model Files* on page 45, we recommend that you decompose your Rose model into controlled units to facilitate parallel development.

We further recommend that where possible, you assign each controlled unit to a team or individual to minimize changes from more than one development view at a time.

More specifically, we recommend that you avoid concurrent changes to Rose diagrams. You can make concurrent changes to the definitions of use cases, classes, and other individual Rose model elements. When you deliver activities, the Model Integrator helps you merge any conflicting changes. The manual, *Guide To Team Development, Rational Rose*, provides more details about working with the Model Integrator.

Changing Files

After you check out files, you have permission to change them. Update your Rose model or source code.

Checking In Files

When you want to keep a record of a checkout's current state, check it in. To check in a file:

- In the ClearCase Explorer, select the file. From the menu, click **Tools > Checkin**.
- From the Rose menu, click **Tools > ClearCase > Check In**. Select the files to check in.

The work you check in from your development view does not become available to other team members until you deliver it.

Delivering an Activity

When you're ready to make an activity available to the project team, prepare your work areas. Then start the deliver operation.

Preparing Work Areas

Take the following steps to prepare your work areas:

- 1 Perform this step if you are delivering code or other text files. If your project manager created a new recommended baseline since you last rebased, rebase your development area. If your project manager has established a policy about rebasing before delivery, then this step is mandatory. For more information about rebasing, see *Rebasing Your Development Stream* on page 60.
- 2 If you are using ClearCase LT or if you are using ClearCase and your integration view is a snapshot view, update the integration view. (The rebase you performed in Step 1 updated your development view.)

When you update, ClearCase copies any files changed by other team members to your view. To update, in the ClearCase Explorer, select a view. From the menu, select **Environment > Update View**.

- 3 In your development view, resolve any hijacked files.

If you are working disconnected from the network and you have not checked out a file, you can *hijack* the file so that you can save your changes. ClearCase notices which files have been hijacked and allows you to check out, then check in your changes after the fact. See ClearCase online Help for more information.

- 4 Perform this step if you are delivering code or other text files. Build and test the work in your development view.

If you are delivering code, it is particularly important to test your changes after you rebase the stream and before you deliver activities.

- 5 Check in all files in the change set of each activity you are delivering.

To deliver an activity, all files associated with that activity must be checked in. In addition, if your project manager has set a policy to disallow deliveries from streams with checkouts, ensure that all files in your view are checked in.

Starting the Deliver Operation

Deliver one or more activities as follows:

- 1 To start the deliver operation:
 - In ClearCase Explorer, right-click the root directory of the development view and click **Deliver from Stream**.
 - In Rose, from the menu, click **Tools > ClearCase > Deliver Stream**.

ClearCase displays the Deliver from Stream Preview.

- 2 In the Preview, select the activity or activities to deliver. The Preview proposes a default view – the integration view associated with your integration stream – to work with. (Remember that you are delivering your development work *to* the integration stream.)
- 3 Click **OK**. ClearCase displays a progress indicator and checks out files to the view you selected in Step 2 and displays a message about merging.
- 4 Read the displayed messages carefully.

Merging

The deliver operation merges the work in your development view with the work in the integration view. It completes most merges for you. If it encounters any conflicts, it prompts you to resolve them. (For more information about merging, see ClearCase online Help.)

Inspecting and Testing

Your integration view contains the merge results. To make sure your delivered work is compatible with the work in the integration view, inspect the files in the integration view. If you are delivering code, this inspection may involve building and testing files. You may also need to perform the following tasks:

- Edit the checked-out versions to resolve any errors you discover.
- Check out and edit additional files.
Note: We recommend that you do not check in any of your changes until you are ready to complete the operation. Checking in complicates efforts to undo the deliver operation.
- Update your integration view to see work other developers have delivered since you started the deliver operation.

Completing the Deliver Operation

When you are satisfied with the changes you made, complete the delivery.

To complete a delivery, click **Complete** in the Delivering to View dialog box. ClearCase checks in your modifications and changes the state of your stream.

Your Project Manager Creates Baselines

At certain points in the development cycle, as dictated by your development process, your project manager creates a new baseline based on the activities that you and your team members have delivered. After testing, your project manager communicates to your team that a new recommended baseline is available.

For more information about steps related to baselining, see *Working with Baselines* on page 63.

Rebasing Your Development Stream

To keep your development work area up-to-date with the activities in your project's recommended baseline, rebase your development stream with each new recommended baseline as it becomes available.

To rebase your development stream:

- 1 Prepare your work area by finding and checking in the work in your development view. You cannot start a rebase operation from a view that contains checkouts.
- 2 Start the rebase operation:
 - In ClearCase Explorer, right-click the root directory of the view and from the shortcut menu, click **Rebase Stream**.
 - In Rose, from the menu, click **Tools > ClearCase > Rebase Stream**.

ClearCase presents the latest recommended baseline. You can choose a different baseline, but it must be newer than the baseline currently in your development stream.

- 3 This step applies most frequently when you are working with code or other text-based files. For any version you modify in your development view, if another team member has modified and delivered a version of the same element, you must merge when you rebase to a baseline that contains the delivered version.

For example:

- a As part of working on an activity, you check in a version of file.c from your development view.
- b Another developer checks in and delivers a new version of file.c.
- c Your project manager adds the new version to the recommended baseline.
- d You rebase your development stream. As part of the rebase operation, you must merge the other developer's version into your development stream.

ClearCase merges all nonconflicting differences. If versions in the baseline contain changes that conflict with the corresponding versions in your development work area, ClearCase prompts you to decide how to resolve conflicting differences.

- 4 This step applies most frequently when you are working with code or other text-based files. After ClearCase reconfigures your work area and completes any needed merges, verify that any undelivered work builds successfully with the versions in the new baseline by initiating a test build in your development view.

In addition to building and testing, you may need to do the following:

- Edit the checked out versions to resolve build errors.
- Check out and edit additional files.

Note: We recommend that you do not check in any of your changes until you are ready to complete the operation. Checking in complicates efforts to undo the rebase operation.

- 5 When you are satisfied with the contents of your development view, complete the rebase: In the Rebase Activities dialog box, click **Complete**.

For More Information

To learn more about the topics described in this chapter, see:

- *Managing Software Projects with ClearCase*, in the ClearCase documentation set.
- *Guide To Team Development, Rational Rose*, in the Rose documentation set.

This chapter describes how an integrator or project manager creates baselines. A baseline represents a configuration of a component. A *recommended* baseline represents a *stable* configuration of a component. After a baseline is created and approved (usually through a test-and-promote process), users rebase their development areas to the most recent recommended baseline, as described in *Developing Software* on page 53.

The steps involved with creating baselines are as follows:

- 1 For requirements and test assets, check in all assets.
- 2 Lock the integration stream.
- 3 For code, build the software components.
- 4 Create the baseline.
- 5 For code, run smoke tests and other tests to confirm the stability of the baseline.
- 6 Promote the baseline.
- 7 Unlock the integration stream.

This chapter also briefly discusses creating a new RequisitePro project or test datastore from a baseline.

Checking In All Assets

As described earlier, a developer working on requirements and test assets does not follow the full UCM model. Instead, the developer works outside of the ClearCase and UCM context. This section describes how the project manager works behind the scenes to support version-controlling these assets.

Displaying the Configure Project Dialog Box

The procedures in this section start in the Configure Project dialog box. To display it:

- 1 Click **Start**, then click **Programs > Rational Suite > Rational Administrator**.
- 2 In the Projects folder, select your Rational project.
- 3 From the Administrator menu, click **File > Configure Project**. The Configure Project dialog box appears.

Checking In Requirements

To check in all requirements in a RequisitePro project:

- 1 Ensure that all users have closed the RequisitePro project.
- 2 On the Configure Project dialog box, in the Requirements Assets area, click **Check In All**. RequisitePro places an exclusive lock on the RequisitePro project and displays a Check In All dialog box.
- 3 In the Check In All dialog box, select or create an activity to associate with the checkin and click **OK**.

RequisitePro performs the following steps:

- Adds any new, unversioned, files to version control.
 - Checks out any files that are currently checked in.
 - Updates all files from the RequisitePro project.
 - Removes any obsolete files from version control.
 - Checks in all files using the activity you specified.
 - Immediately checks out all files.
- 4 When the checkin operation completes, close the Configure Project dialog box.
 - 5 RequisitePro users can now resume work in RequisitePro.

Checking In Test Assets

The process for checking in all test assets is similar to the one for checking in all requirements:

- 1 On the Configure Project dialog box, in the Test Assets area, click **Check In All**. A Check In All dialog box appears.
- 2 In the Check In All dialog box, select or create an activity to associate with the checkin and click **OK**.

Rational Test performs the following steps:

- Adds any new, unversioned, files to version control.
 - Checks out any files that are currently checked in.
 - Updates all files from the test datastore.
 - Removes any obsolete files from version control.
 - Checks in all files using the activity you specified.
 - Immediately checks out all files.
- 3 When the checkin operation completes, close the Configure Project dialog box.

Starting the ClearCase Project Explorer

You use the ClearCase Project Explorer to perform many of the steps in the following sections. To start the Project Explorer:

- From the Rational Administrator menu, click **Tools > Rational ClearCase Project Explorer**.
- From the desktop, click **Start**, then click **Programs > Rational ClearCase > Project Explorer**.

Locking the Integration Stream

Before you create the baseline, always lock the integration stream to prevent concurrent activity in the stream.

- 1 In the ClearCase Project Explorer, select the integration stream.
- 2 Click **File > Properties** to display the integration stream's property sheet.
- 3 On the **Lock** tab, select **Locked**, then click **OK**.

Creating the Baseline

Create the baseline as follows:

- 1 In the ClearCase Project Explorer, select the project's integration stream.
- 2 Click **Tools > Make Baseline**. The Make Baseline dialog box appears with default values supplied for most of the fields.
- 3 In the Make Baseline dialog box, make changes as appropriate, then click **OK**.
- 4 If you see a message saying that the integration view contains elements that are currently checked out, take one of the following actions:
 - If your baseline includes requirements or test assets, ClearCase always displays this message. The Check In All operation that you performed earlier checks in all files, then immediately checks them out. It is safe to ignore this message; click **OK** to continue.
 - If your baseline contains other types of artifacts, click **Find Check Outs** and take appropriate actions to resolve the checkouts if you wish. (ClearCase online Help can provide guidelines for resolving checkouts.) When you are ready, you can continue with creating the baseline.

Unlocking the Integration Stream

Unlock the integration stream to make it available for changes:

- 1 In the ClearCase Project Explorer, select the integration stream.
- 2 Click **File > Properties** to display the integration stream's property sheet.
- 3 On the **Lock** tab, select **Unlocked**, then click **OK**.

Building and Testing the Software Components

If your baseline includes code, we recommend that you build and test the project's executable files to confirm the stability of the new baseline. We recommend that your testing team test the baseline on a designated testing development stream. This technique frees the integration stream so that developers can continue to deliver activities.

When the builds and tests succeed, the project manager can promote the baseline and users can rebase their development streams.

Promoting the Baseline and Working with Promotion Levels

Each component has at least one baseline (but usually many more). Only one baseline at a time can be a component's recommended baseline.

The recommended baseline is the most recent baseline at or above a certain promotion level. By default, ClearCase provides an ordered set of promotion levels (*rejected, initial, built, tested, released*) and designates the promotion level at or above which a baseline will be recommended. Each promotion level represents the degree of confidence in the stability, quality, or completeness of the components in the baseline.

The rest of this section describes how to:

- Promote all baselines to the recommended promotion level that designates a recommended baseline.
- Change the promotion level of a specific baseline.
- For a project, declare which promotion level designates a recommended baseline.
- For a project, change the list of promotion levels.

Promoting All Baselines to the Recommended Level

To promote all baselines to the recommended level:

- 1 In the ClearCase Project Explorer, select the integration stream, then click **Tools > Recommend Baselines**. The Recommend Baselines dialog box lists the components with baselines below the recommended level.
- 2 Click **OK** to promote the baselines of the listed components to the recommended level.

Your users can now rebase their development stream, as described in *Rebasing Your Development Stream* on page 60.

Changing One Component's Baseline Promotion Level

You may want to change a baseline's promotion level to an intermediate level, for example, to note that the baseline has been successfully built but not yet tested. Or, you may want to change one component's baseline promotion level without changing others.

To change a component's baseline promotion level:

- 1 In the ClearCase Project Explorer, select the integration stream, then click **File > Properties**.
- 2 On the **Baselines** tab of the properties sheet, select a component. The component's baselines appear in the Baselines area.
- 3 From the Baselines area, select one of the baselines, then click **Properties**.
- 4 On the **General** tab of the baseline's property sheet, select a new value from the Promotion Level list.

You can use this technique to promote or demote a baseline. For example, if, during testing, a major bug is discovered, you may want to set the Promotion level to *rejected* to prevent anyone from rebasing to this baseline.

Your users can now rebase their development stream, as described in *Rebasing Your Development Stream* on page 60.

Declaring the Recommended Promotion Level

You can designate the recommended promotion level for each UCM project. For example, when you are first starting work on a project, the *initial* promotion level might be sufficient for your team. As soon as testing gets underway, however, you probably don't want to recommend a baseline unless it's at the *tested* promotion level. When your project goes to internal deployment, you might want to change the recommended promotion level again, to *released*.

To declare the recommended promotion level:

- 1 In the ClearCase Project Explorer, select a project and click **File > Properties**. (*Hint*: when you select a project in the left pane, the right pane displays streams associated with the project.)
- 2 On the Policy page, select a value from the list, **Recommend baselines at or above this promotion level**.

Changing the List of Promotion Levels

ClearCase by default provides an ordered set of promotion levels. To change them:

- 1 In the ClearCase Project Explorer, select the top-most folder and click **Tools > Define Promotion Level**. (*Hint*: when you select the folder in the left pane, the right pane displays a Components folder and the projects associated with the folder.)
- 2 Use the Define Promotion Levels dialog box to add, remove, or change the order of the promotion levels.

Creating a New RequisitePro Project or Test Datastore from a Baseline

From the baseline of a RequisitePro project, you can create a new RequisitePro project. From the baseline of a test datastore, you can create a new test datastore. For example, you may want to start the next release of a project based on a stable configuration of the previous release. Details of these processes are described in *Using the Rational Administrator*.

For More Information

To learn more about the topics described in this chapter, see:

- *Using the Rational Administrator*, in the Rational Suite documentation set.

This chapter briefly describes topics to consider if your environment is set up differently from what is described in this manual. It covers these areas:

- Using UCM with or without ClearQuest
- Using UCM with ClearCase instead of ClearCase LT
- Using ClearCase or ClearCase LT without using UCM

Using UCM With or Without ClearQuest

This manual assumes that you use UCM with ClearQuest. It is also possible to use UCM without ClearQuest. This section describes the differences between using UCM with ClearQuest and using UCM without ClearQuest.

Without ClearQuest, UCM allows you to create activities (or use existing ones) when you check out and track which activities have been delivered. You can also create baselines of projects based on the activities that have already been delivered.

ClearQuest extends UCM by adding project management features. With ClearQuest, project managers can:

- Assign activities to developers. (ClearQuest maintains “to-do” lists, containing the activities assigned to each developer.)
- Use queries to see which developers are working on which activities, and to see what state each activity is in (completed, in process, not started, and so on).
- Create reports and charts to track and illustrate the progress of the project.

Your project might not use ClearQuest. For example, your project may already use another change request management system and may not be able to migrate to ClearQuest right now. However, as your project’s needs grow more complex, you can benefit from the integrations and project management capabilities that ClearQuest offers.

Using UCM With ClearCase Instead of ClearCase LT

ClearCase LT is included in all editions of Rational Suite. As part of a typical install, ClearCase LT is placed on your system as long as you have not previously installed ClearCase.

ClearCase LT is a subset of ClearCase and provides for more simplified setup, use, and maintenance as compared with ClearCase. ClearCase LT is designed for relatively small groups that are geographically co-located and that do not need to share files or artifacts with other groups. If your group's needs change, there is an easy upgrade path from ClearCase LT to ClearCase.

Both ClearCase LT and ClearCase fully support UCM. For the features they share, the user interface is exactly the same for both tools.

The following items describe key features provided by ClearCase that are not available in ClearCase LT:

- Multiple servers

With ClearCase, you can set up multiple servers. By contrast, for each group of ClearCase LT users, there can be exactly one server. A ClearCase server hosts VOBs and other administrative files and directories.

- MultiSite support

ClearCase supports the use of Rational ClearCase MultiSite. MultiSite extends ClearCase by supporting parallel software development and software reuse across geographically distributed project teams. MultiSite enables developers at different locations to use the same VOB. Each site has its own copy, or *replica*, of the VOB. Sites propagate changes to other sites using an automated or manual synchronization process.

- Dynamic and snapshot views

ClearCase LT provides one type of view, a *snapshot* view, which uses a copy method to give you access to files in a VOB. With a snapshot view, you decide which components you want to work with, then copy files from the VOB to your system. When you want to see changes from other developers, you need to update your view to contain the latest contents of the VOB.

ClearCase provides snapshot views, too, but in addition, provides *dynamic* views, which are file-system based. A dynamic view does not require you to copy files to your system. You designate which VOBs to use, and using the dynamic view, you can work with any file in those VOBs. When files change, a dynamic view updates automatically; you do not need to update the view yourself.

- Code-building features

ClearCase provides build tools, based on make, that are not available in ClearCase LT. The make program was developed for UNIX systems and has been ported to other systems. The ClearCase build tools provide compatibility with make variants and include powerful enhancements.

Using ClearCase or ClearCase LT Without Using UCM

UCM provides an out-of-the-box, configurable process to use with ClearCase LT or ClearCase. We recommend that you use UCM unless your group has a special need that requires a non-UCM process.

If you use base ClearCase, that is, ClearCase without UCM, you need to define a configuration management process for your group. ClearCase provides a wide range of graphical tools, a command language, and a scripting language to support your group's end users, administrators, build engineers, and project managers. For more information about setting up a base ClearCase environment and working with it, see *Managing Software Projects with ClearCase*.

For More Information

To learn more about the topics described in this chapter, see:

- *Managing Software Projects with ClearCase*, in the ClearCase documentation set.

Index

A

- activity 13, 17
 - and change requests 20
 - and ClearQuest 29, 71
 - delivering 57
 - setting 56
 - with or without ClearQuest 55
 - working on 55
- artifact 13

B

- baseline 15, 18, 63
 - creating 66
 - promoting 67
 - promotion level 67
 - recommended 67
 - using to create RequisitePro project 69
 - using to create test datastore 69
- branch (ClearCase) 19

C

- change request
 - and activities 20
 - and Rational project 44
 - and UCM 44
- change set 15
- check in all operation 63
- checking in (ClearCase) 20, 57
- checking out (ClearCase) 20, 56
- ClearCase 13, 18
 - and MultiSite 72
 - branch 19
 - checking in 20, 57
 - checking out 20, 56
 - dynamic view 72
 - element 18

- hijacking a file 57
- merging versions 58
- replica 72
- snapshot view 72
- updating a view 57
- version tree 18
- view 20
 - without UCM 72
- ClearCase Explorer 38, 54, 57, 58, 60
- ClearCase LT
 - client 25
 - server 25
 - server and ClearQuest 30
 - without UCM 72
- ClearCase LT Server Setup Wizard 26
- ClearCase Project Explorer 47, 65, 66, 68
- ClearQuest 13
 - and activities 29
 - and ClearCase LT server 30
 - and Rational project 44
 - and UCM 71
 - creating user database 31
 - schema 21
 - schema repository 21, 30
 - schemas and UCM 31
 - setting up environment 29
 - setting up schema repository 30
 - state 20
 - user database 20, 30
- ClearQuest Maintenance tool 30
- client (ClearCase LT) 25
- code, developing 53
- component 17, 27
- Configure Project dialog box 40, 42, 44, 45, 47, 64, 65
- controlled units (Rose) 45
- Create RequisitePro Project dialog box 42
- Create Test Datastore tool 44

D

delivering an activity 15, 57
development stream 15, 17
dynamic view (ClearCase) 72

E

element (ClearCase) 18

H

hijacking a file (ClearCase) 57

I

integration stream 15, 17
locking 65
unlocking 66

J

Join Project wizard 54
joining a project 15, 54

L

load rules and ClearCase view 38
locking a stream 65

M

Managing Rose Model Files 45
merging versions (ClearCase) 19, 58
MultiSite, and ClearCase 72

N

New Project Wizard 40

P

project
Rational 39
RequisitePro 41
UCM 17, 26
promoting a baseline 67
promotion level 67
PVOB (UCM) 20, 26

R

Rational Administrator 39
Rational project 15
 associating with change requests 44
 associating with requirements 41
 associating with test assets 43
 associating with visual models 45
 setting up 39
Rational Test
 and Rational project 43
 and UCM 16
Rational Unified Process 13
rebasng a stream (UCM) 16, 60
recommended baseline 67
replica (ClearCase) 72
requirements
 and Rational project 41
 and UCM 41
 check in all 64
 creating RequisitePro project from
 baseline 69
 developing 53
RequisitePro
 and Rational project 41
 and UCM 16
 project 41
Rose
 and Rational project 45
 and UCM 16
 controlled units 45
 developing models 53
 guidelines 45, 56

S

- schema (ClearQuest) 21
- schema repository (ClearQuest) 21, 30
- server (ClearCase LT) 25
- setting an activity 56
- snapshot view (ClearCase) 72
- state (ClearQuest) 20
- stream (UCM) 17, 20
 - locking 65
 - unlocking 66

T

- test assets
 - and Rational project 43
 - and UCM 43
 - check in all 65
 - creating datastore from baseline 69
 - developing 53
- to-do lists, ClearQuest 71

U

- UCM 13
 - and ClearQuest schemas 31
 - overview of using 14
 - project 17, 26
 - PVOB 20, 26
 - rebasing a stream 60
 - setting up environment 24
 - stream 20
 - with or without ClearQuest 71
- Unified Change Management. *See UCM.*
- unlocking a stream 66
- updating a view (ClearCase) 39, 57
- user database (ClearQuest) 20, 30
 - creating 31

V

- version tree (ClearCase) 18
- view (ClearCase) 20
 - dynamic or snapshot 72
 - load rules 38
- visual models and UCM 45
- VOB 18
- VOB Creation wizard 37

