

Using Unified Change Management with Rational Suite

VERSION: 2001.03.00

PART NUMBER: 800-023724-000

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

Rational[®]
the e-development company™

IMPORTANT NOTICE

Copyright Notice

Copyright © 1998, 1999, 2000 Rational Software Corporation. All rights reserved.

Trademarks

Rational, the Rational logo, ClearCase, PureCoverage, PerformanceStudio, Purify, Quantify, Rational Rose, and SoDA are trademarks or registered trademarks of Rational Software Corporation in the United States and in other countries. All other names are used for identification purposes only and are trademarks or registered trademarks of their respective companies.

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

Microsoft, MS, ActiveX, BackOffice, Developer Studio, Visual Basic, Visual C++, Visual InterDev, Visual J++, Visual Studio, Win32, Windows, and Windows NT are trademarks or registered trademarks of Microsoft Corporation.

Rational Purify is licensed under Sun Microsystem's U.S. Pat. No 5,404,499.

Oracle, Oracle7, and Oracle 8 are trademarks or registered trademarks of Oracle Corporation.

Sybase and SQL Anywhere are trademarks or registered trademarks of Sybase Corporation.

U.S. Government Rights

Use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in the applicable Rational License Agreement and in DFARS 227.7202-1(a) and 227.7202-3(a) (1995), DFARS 252.227-7013(c)(1)(ii) (Oct 1988), FAR 12.212(a) 1995, FAR 52.227-19, or FAR 52.227-14, as applicable.

Patent

U.S. Patent Nos. 5,193,180 and 5,335,344 and 5,535,329 and 5,835,701. Additional patents pending.

Warranty Disclaimer

This document and its associated software may be used as stated in the underlying license agreement, and, except as explicitly stated otherwise in such 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	7
Audience.....	7
Other Resources.....	7
Rational Suite Documentation Roadmap.....	8
Contacting Rational Technical Support.....	9
1 What is UCM?	11
The Building Blocks.....	11
Why Use UCM?.....	12
Typical Way to Use UCM.....	12
A Note About Tools.....	14
For More Information.....	14
2 UCM Concepts	15
UCM Basics.....	15
Project.....	15
Component.....	15
Activity.....	15
Stream.....	15
Baseline.....	16
ClearCase Basics.....	16
VOBs and Their Contents.....	16
Views and Streams.....	18
Using ClearCase.....	18
Behind the Scenes.....	18
ClearQuest Basics.....	18
For More Information.....	19
3 Setting Up UCM	21
Setting Up ClearQuest.....	21
Installing ClearQuest.....	22
Setting Up a Schema Repository.....	22
Setting Up a User Database.....	23
Guidelines for Choosing a Schema.....	23
Configuring a User Database.....	23

Setting Up a ClearCase LT Server	24
Installing and Configuring the ClearCase LT Server.	24
Starting the ClearCase LT Server Setup Wizard	24
Working with the Server Setup Wizard.	24
Starting the ClearCase Project Explorer.	25
Creating an Integration View	26
Associating the ClearQuest Database with ClearCase LT	26
Adding another Component to the Project	27
Creating the Component	27
Adding the Component's Baseline to the Integration Stream.	27
Updating Development and Integration Views	28
Installing ClearCase LT Clients.	29
Setting Up a Rational Project	29
Creating a Rational Project.	29
Configuring a Rational Project	30
Associating Requirements	31
Associating Test Assets	32
Associating Change Requests	33
Associating Rose Models.	33
Creating a Baseline	35
For More Information.	35
4 Developing Software	37
Developing Requirements and Test Assets.	37
Developing Rose Visual Models and Code.	37
Joining a Project	38
Working on an Activity	39
Setting an Activity: With ClearQuest	39
Setting an Activity: Without ClearQuest.	39
Checking Out Files.	40
Changing Files	40
Checking In Files	40
Delivering an Activity	41
Preparing Work Areas	41
Starting the Deliver Operation	42
Merging	42
Inspecting and Testing	42
Completing the Deliver Operation	43

Your Project Manager Creates Baselines	43
Rebasing Your Development Stream	43
For More Information	44
5 Working with Baselines	45
Checking In all Assets	45
Displaying the Configure Project Dialog Box	46
Checking in Requirements	46
Checking in Test Assets	47
Starting the ClearCase Project Explorer	47
Locking the Integration Stream	47
Creating the Baseline	48
Unlocking the Integration Stream	48
Building and Testing the Software Components	48
Promoting the Baseline and Working with Promotion Levels	49
Promoting All Baselines to the Recommended Level	49
Changing One Component's Baseline Promotion Level	49
Declaring the Recommended Promotion Level	50
Changing the List of Promotion Levels	50
Creating a New RequisitePro Project or Test Datastore from a Baseline	51
For More Information	51
6 Variations in Your Environment	53
Using UCM With or Without ClearQuest	53
Using UCM With ClearCase instead of ClearCase LT	54
Using ClearCase or ClearCase LT Without Using UCM	55
For More Information	55
Index	57

Preface

This manual shows you how to 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 an out-of-the-box 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 life cycle. Rational Suite's unparalleled level of integration improves communication both within teams and across team boundaries, reducing development time and improving software quality.

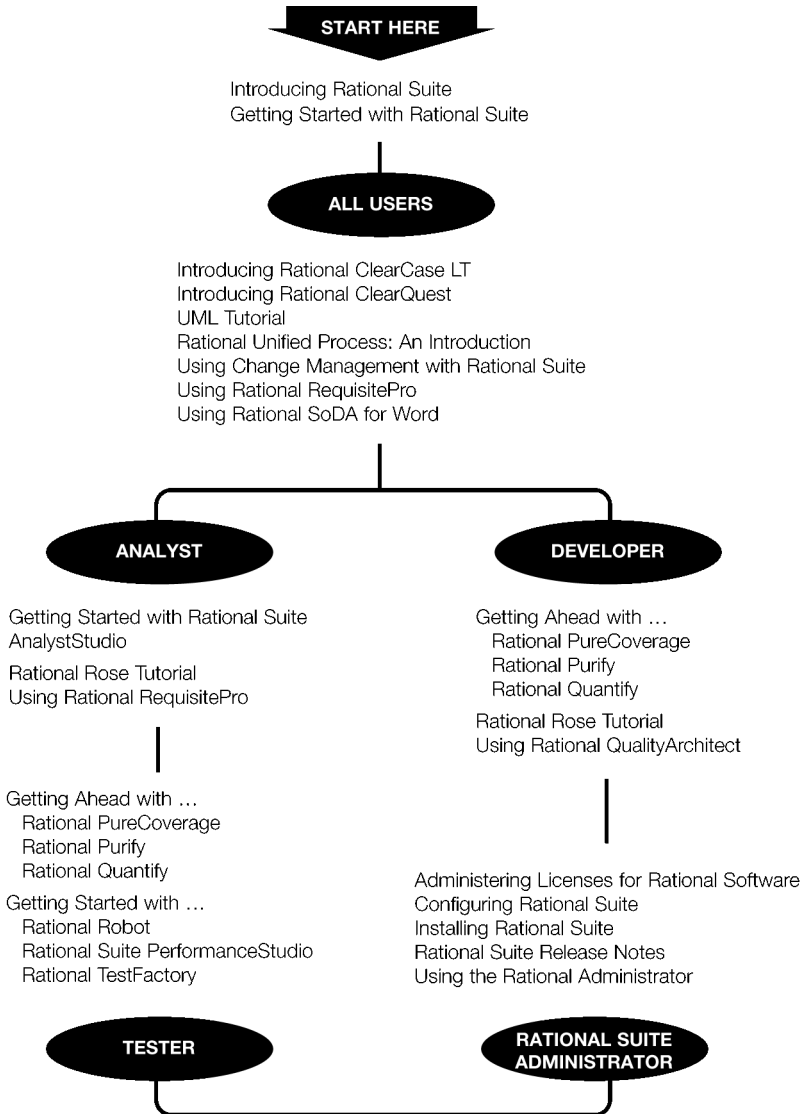
Audience

This guide 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

- Online Help is available for Rational Suite.
From a Suite tool, select an option from the **Help** menu.
- 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>.

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, telephone number, and company name
- Your computer's make and model
- Your operating system and version number
- Product release number and serial 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 book 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 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 the artifacts produced by a software project.

Why Use UCM?

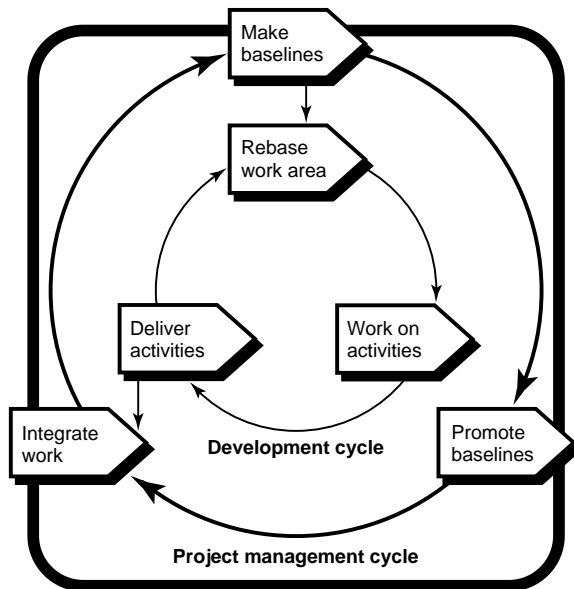
Rational customers and Rational employees have successfully used ClearCase and ClearQuest for years. During that time, common patterns have emerged regarding how users set up and use the tools. Based on our experience and observations, we've created UCM, a process that helps you get started with ClearCase 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 has been 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, please see *Setting Up UCM* on page 21.
- 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, please see *Developing Software* on page 37). 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, please see *Working with Baselines* on page 45.
- 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 book 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 53 discusses these alternative environments. However, even if you work in a different environment, most of the information in this manual is still relevant.

The Rational Suite tools that are integrated with UCM (and that are discussed in this manual) are:

- 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, please 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 11, 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 18.

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.

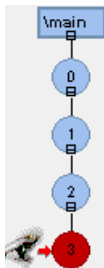
ClearCase Basics

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

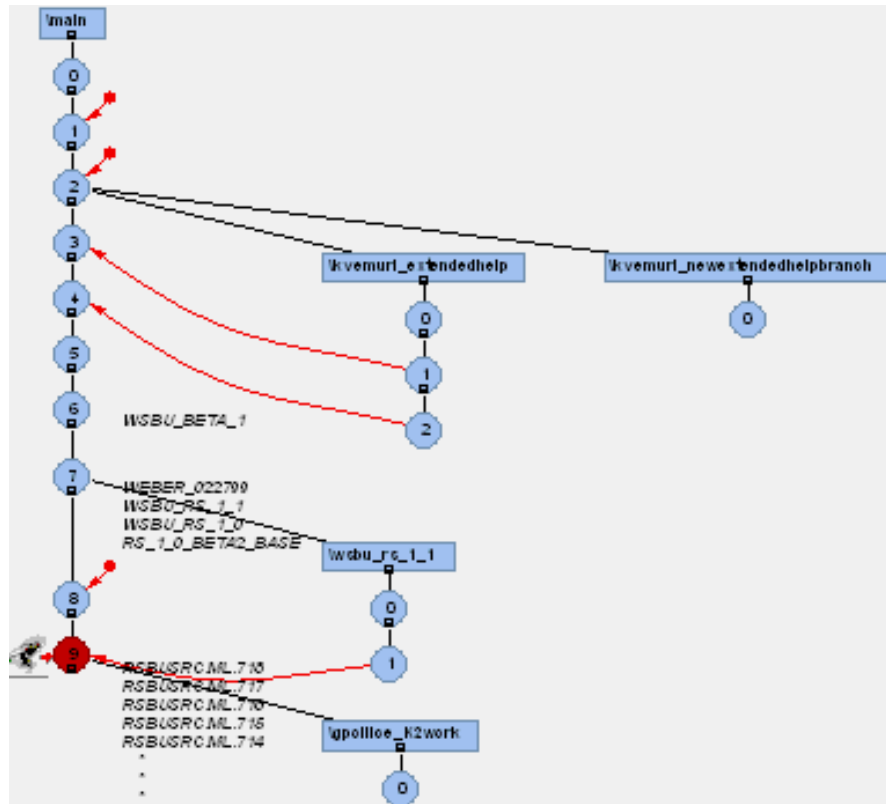
ClearCase stores information in a database, called a *VOB* (for versioned object base). A VOB contains *elements*; each element stores a project directory or file. For each element, ClearCase keeps track of each version you create. So, for example, in a fairly simple development environment, one element might have versions as shown in the following *version tree*. (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've reached 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 11, 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 machines. For more information, see *Setting Up UCM* on page 21.

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, please see:

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

This chapter describes how a project manager sets up a UCM environment that works with Rational Suite. To set up the environment, follow these steps in this order:

- 1 Optionally, install and configure Rational ClearQuest.
- 2 Install and configure a Rational ClearCase LT server.
- 3 Install Rational Suite and create a Rational project.

Note: This chapter describes placing all tools on one server. This configuration works for small teams. As your team grows, and as you place greater demands on ClearCase and ClearQuest, performance may degrade. In this case, we recommend a configuration with dedicated servers for ClearCase and ClearQuest.

Setting Up ClearQuest

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

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

- 1 Install ClearQuest.
- 2 Set up a schema repository.
- 3 Set up a user database. When you configure the ClearCase LT server, you will be prompted to provide the name of this user database.

Installing ClearQuest

To install ClearQuest, follow these high-level steps:

- 1 Set up the database product that will store ClearQuest information, as described in *Installing Rational ClearQuest*.
- 2 Choose the machine on which you will create ClearQuest databases. Select a highly available machine that is backed up regularly. Install ClearQuest on this machine, following these tips:
 - You can install ClearQuest either as a standalone product or as part of Rational Suite.
 - This tip applies if you install Rational Suite on the same machine as the machine where the ClearCase LT server will reside. On the Setup Configuration page, choose **Custom/Full** and clear the checkbox next to Rational ClearCase LT. (The Suite includes Rational ClearCase LT client; if ClearCase LT client is installed, you cannot later install ClearCase LT server.)
 - On the Software Use page, select **Administrator**.

To learn more about topics related to installing ClearQuest, read *Installing Rational ClearQuest*. This book contains information about setting up database products, installing ClearQuest, and creating ClearQuest databases.

Setting Up a Schema Repository

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

Once 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 by clicking **Start**, and then choosing **Programs > Rational Suite > Rational ClearQuest > Rational ClearQuest Maintenance Tool**.
- 2 On the first page, choose **Create a new schema repository**.
- 3 On the Schema Repository Properties page, choose the type of database and supply information related to that database type. *Installing Rational ClearQuest* contains details about how to enter the information on this page.

- 4 On the Create Sample Database page, you have the option of creating a sample user database for testing purposes. If you do create the sample database, choose the schema to use with the database. If you plan to associate the sample database with a UCM project, follow the guidelines for choosing a schema described under *Setting Up a User Database*.
- 5 If you choose to create a sample database, enter information about the database on the Sample User Database Properties page.
- 6 On the Save Current Setting for Later Use page, you can optionally choose to save the information you've entered in a profile file. If you do 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 26) determines which schema will be used with the ClearCase LT server.

Configuring a User Database

The manuals, *Managing Software Projects with ClearCase* and *Administering Rational ClearQuest* describe how to complete the configuration process by setting up state types and state transitions.

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* machines to get access to the project information and other files that reside on the server.

The remainder of this section gives an overview of setting up the ClearCase LT server.

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 Select Installation Option page, select **Install Rational ClearCase LT Server**.

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. Once you log in as the correct user, start the Server Setup Wizard: click **Start** and then click **Programs > Rational ClearCase LT Server > ClearCase Server Setup Wizard**.

In more detail, 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.

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 will help 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 in another source control system that you'd like to place under ClearCase control, 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 *Setting Up a Rational Project* on page 29 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.
 - 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.

Starting the ClearCase Project Explorer

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

Click **Start**, then click **Programs > Rational ClearCase LT Server > Rational ClearCase Project Explorer**.

Creating an Integration View

The integration view, together with the integration stream, define the common workspace accessible to all team members. To create the integration view:

- 1 In the ClearCase Project Explorer, in the tree-browser, open the **projects** folder, then open your project by double-clicking it. An **Integration stream** icon appears below the project name.
- 2 Right-click the **Integration stream** icon, and from the shortcut menu, select **Create View**. The View Creation Wizard appears.
- 3 On the wizard's first page, specify a directory where there is enough disk space to contain the project's files.
- 4 Read the confirmation page and click **OK**.

ClearCase prompts you to load elements into your view. ClearCase LT works with snapshot views which copy files from the VOB onto your disk.

- 5 Select the components associated with this project and add them to the Selected Elements list. You may need to select **Show All VOBs** to see a complete list of available components.

ClearCase updates your view with the directories and files currently in the components you selected.

- 6 Leave the ClearCase Project Explorer open for the next task.

Associating the ClearQuest Database with ClearCase LT

Associate the ClearQuest user database you created with the ClearCase project:

- 1 In the left pane of the ClearCase Project Explorer, navigate to your project (under the **projects** folder). Right-click the project. From the shortcut menu, select **Properties**.
- 2 On the project properties sheet, on the ClearQuest tab:
 - a Select the **Project is ClearQuest-enabled** checkbox.
 - b If you see a message that this project is the first to be UCM-enabled, read it carefully before proceeding.
 - c Select the ClearQuest user database you plan to use.
 - d Select the ClearQuest policies you want to enforce on your project. The online "What's This" Help for this property sheet explains each policy. For example, to close an activity after delivering it, select **Do ClearQuest Action After Delivery**.
- 3 Click **OK**.

Adding another Component to the Project

You may want to associate more than one component with a 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 ClearCase LT Server > 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 Leave **Create VOB as UCM component** selected.
 - c 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:

- Updating the integration stream
- Updating 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 In the ClearCase Project Explorer, in the tree browser, open the **projects** folder, then open your project by double-clicking it. An **Integration stream** icon appears below the project name.
- 2 Right-click the **Integration stream** icon and from the shortcut menu, select **Properties**. The Integration Stream Properties sheet appears.

- 3 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.
- 4 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.
- 5 Click **OK** to close the Integration Stream Properties sheet, then click **OK** to close the Confirm message box.
- 6 ClearCase offers to rebase your stream. Click **OK**.
- 7 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 ClearCase LT > Rational 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.

Installing ClearCase LT Clients

Users typically install ClearCase LT clients as part of installing a Rational Suite edition. The installation prompts the user to supply the name of the ClearCase LT server. Alternately, the user can specify the server name after installation by using the ClearCase control panel applet. We recommend that you complete the steps in this chapter before instructing users to install Rational Suite.

Setting Up a Rational Project

To set up a Rational project, follow these high-level steps:

- 1 Install Rational Suite.

Install Rational Suite on your system by following the instructions outlined in *Installing Rational Suite*.

- 2 Create and configure the Rational project.

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

These steps are described in the remainder of this section.

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, please see *Using the Rational Administrator*.

To create a Rational project:

- 1 Create a share name on the machine where you want to place the project. Choose an area that is not under version control. In other words, the area should not be part of a VOB.
- 2 Start the Rational Administrator. From the **Start** menu, click **Programs > Rational Suite > Rational Administrator**.
- 3 Start the New Project Wizard by clicking **File > New Project** on the Administrator menu.
- 4 On the wizard's first page:
 - a Supply a name for your project. 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 Step 1. A UNC pathname has the format `\\machine\share\subdirectory`. Make sure that you specify a path that is at least in a subdirectory of the share.
 - c Make sure that you select **UCM enabled**.
- 5 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.
- 6 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.
- 7 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**.

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.

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.

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, specify an area in your integration view where you'd like the requirements to reside. For example, you might specify `disk:\view\VOBrequirements`.
 - c In the Database field, specify the type of database you plan to use with your RequisitePro project.
 - d Optionally, provide a comment describing the RequisitePro project.
 - e Click **OK**. RequisitePro creates the project and adds it to version control.

Associating Test Assets

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.

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, specify an area in your integration view where you'd like the tests to reside. For example, you might specify `disk:\view\VOB\tests`.
 - b Select initialization options as appropriate.
 - c Click **OK**. Rational Test creates a test datastore and adds it to version control.

Associating Change Requests

If you've 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 earlier in this chapter.

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, please read about breaking a model into controlled units in *Rational Rose Guide to Team Development*.

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.

- 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

Once you have set up the Rational project and placed 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 45.

For More Information

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

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

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), please 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, please see *Working with Baselines* on page 45.

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 11 for an overview of the UCM model.) This section describes the following work:

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

To perform the work described in this section, you typically use the Rational ClearCase Explorer or a menu item in your code development environment. To start the ClearCase Explorer from your desktop, click **Start**, then click **Programs > Rational ClearCase LT Server > 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 **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 will be identical, but they will 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 report, 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, select **Tools > Checkout**.
- From the Rose menu, select **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 33, 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 classes. When you deliver activities, the Model Integrator helps you merge any conflicting changes to classes. The manual, *Rational Rose Guide to Team Development*, provides more details about working with the Model Integrator.

Changing Files

Once 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, select **Tools > Checkin**.
- From the Rose menu, select **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 has created a new recommended baseline since you last rebased, rebase your development area. For more information about rebasing, please see *Rebasing Your Development Stream* on page 43.
- 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 any files that are still checked out.

To deliver an activity, all files associated with that activity must be 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 select **Deliver from Stream**.
 - In Rose, from the menu, select **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 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, please 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 have 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, please see *Working with Baselines* on page 45.

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, select **Rebase Stream**.
 - In Rose, from the menu, select **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, please see:

- *Managing Software Projects with ClearCase*, in the ClearCase documentation set.
- *Rational Rose Guide to Team Development*, 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. Once 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 37.

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 the **Start** button, then click **Programs > Rational Suite > Rational Administrator**.
- 2 In the Projects folder, select your Rational project.
- 3 From the Administrator menu, select **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 perform many of the following steps in the ClearCase Project Explorer. 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.

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 43.

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 43.

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, please see:

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

This chapter briefly describes topics to consider if your environment is set up differently from what is described in this book. 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 book 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, please see *Managing Software Projects with ClearCase*.

For More Information

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

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

Index

A

- activity 11, 15
 - and change requests 18
 - and ClearQuest 21, 53
 - delivering 41
 - setting 39
 - with or without ClearQuest 39
 - working on 39
- artifact 11

B

- baseline 13, 16, 45
 - creating 48
 - promoting 49
 - promotion level 49
 - recommended 49
 - using to create RequisitePro project 51
 - using to create test datastore 51
- branch (ClearCase) 17

C

- change request
 - and activities 18
 - and Rational project 33
 - and UCM 33
- change set 13
- check in all operation 45
- checking in (ClearCase) 18, 40
- checking out (ClearCase) 18, 40
- ClearCase 11, 16
 - and MultiSite 54
 - branch 17
 - checking in 18, 40
 - checking out 18, 40
 - dynamic view 54

- element 16
- hijacking a file 41
- merging versions 42
- replica 54
- snapshot view 54
- updating a view 41
- version tree 16
- view 18
 - without UCM 54
- ClearCase Explorer 28, 38, 41, 42, 43
- ClearCase LT
 - client 24
 - server 24
 - server and ClearQuest 22
 - server setup 24
 - without UCM 54
- ClearCase LT Server Setup Wizard 24
- ClearCase Project Explorer 25, 26, 27, 35, 47, 48, 49
- ClearQuest 11
 - and activities 21
 - and ClearCase LT server 22
 - and Rational project 33
 - and UCM 53
 - creating user database 23
 - installing 22
 - schema 19
 - schema repository 19, 22
 - schemas and UCM 23
 - setting up environment 21
 - setting up schema repository 22
 - state 18
 - user database 18, 22
- ClearQuest Maintenance tool 22
- client (ClearCase LT) 24
- code, developing 37
- component 15, 25
- Configure Project dialog box 30, 32, 33, 34, 46, 47

controlled units (Rose) 33
Create RequisitePro Project dialog box 32
Create Test Datastore tool 33

D

delivering an activity 13, 41
development stream 13, 15
dynamic view (ClearCase) 54

E

element (ClearCase) 16

H

hijacking a file (ClearCase) 41

I

installing ClearQuest 22
integration stream 13, 15
 locking 47
 unlocking 48

J

Join Project wizard 38
joining a project 13, 38

L

load rules and ClearCase view 28
locking a stream 47

M

Managing Rose Model Files 33
merging versions (ClearCase) 17, 42
MultiSite, and ClearCase 54

N

New Project Wizard 30

P

project
 Rational 29
 RequisitePro 31
 UCM 15, 24
promoting a baseline 49
promotion level 49
PVOB (UCM) 18, 24

R

Rational Administrator 29
Rational project 13
 associating with change requests 33
 associating with requirements 31
 associating with test assets 32
 associating with visual models 33
 setting up 29
Rational Test
 and Rational project 32
 and UCM 14
Rational Unified Process 11
rebasng a stream (UCM) 14, 43
recommended baseline 49
replica (ClearCase) 54
requirements
 and Rational project 31
 and UCM 31
 check in all 46
 creating RequisitePro project from
 baseline 51
 developing 37
RequisitePro
 and Rational project 31
 and UCM 14
 project 31

Rose

- and Rational project 33
- and UCM 14
- controlled units 33
- developing models 37
- guidelines 33, 40

S

- schema (ClearQuest) 19
- schema repository (ClearQuest) 19, 22
- server (ClearCase LT) 24
- setting an activity 39
- snapshot view (ClearCase) 54
- state (ClearQuest) 18
- stream (UCM) 15, 18
 - locking 47
 - unlocking 48

T

- test assets
 - and Rational project 32
 - and UCM 32
 - check in all 47
 - creating datastore from baseline 51
 - developing 37
- to-do lists, ClearQuest 53

U

- UCM 11
 - and ClearQuest schemas 23
 - overview of using 12
 - project 15, 24
 - PVOB 18, 24
 - rebasing a stream 43
 - setting up environment 21
 - stream 18
 - with or without ClearQuest 53
- Unified Change Management. *See UCM.*
- unlocking a stream 48
- updating a view (ClearCase) 29, 41
- user database (ClearQuest) 18, 22
 - creating 23

V

- version tree (ClearCase) 16
- view (ClearCase) 18
 - dynamic or snapshot 54
 - load rules 28
- View Creation Wizard 26
- visual models and UCM 33
- VOB 16
- VOB Creation wizard 27

