Rational® ClearCase®

Developing Software

VERSION: 2003.06.00 AND LATER

PART NUMBER: 800-026156-000

WINDOWS EDITION

Rational®
the software development company

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

Legal Notices

Copyright ©1992-2003, Rational Software Corporation. All Rights Reserved.
Part Number: 800-026156-000
Version Number: 2003.06.00

This manual (the "Work") is protected under the copyright laws of the United States and/or other jurisdictions, as well as various international treaties. Any reproduction or distribution of the Work is expressly prohibited without the prior written consent of Rational Software Corporation.

The Work is furnished under a license and may be used or copied only in accordance with the terms of that license. Unless specifically allowed under the license, this manual or copies of it may not be provided or otherwise made available to any other person. No title to or ownership of the manual is transferred. Read the license agreement for complete terms.

Rational Software Corporation, Rational, Rational Suite, Rational Suite ContentStudio, Rational Apex, Rational Process Workbench, Rational Rose, Rational Summit, Rational Unified process, Rational Visual Test, AnalystStudio, ClearCase, ClearCase Attache, ClearCase MultiSite, ClearDDTS, ClearGuide, ClearQuest, PerformanceStudio, PureCoverage, Purify, Quantify, Requisite, RequisitePro, RUP, SiteCheck, SiteLoad, SoDa, TestFactory, TestFoundation, TestMate and TestStudio are registered trademarks of Rational Software Corporation in the United States and are trademarks or registered trademarks in other countries. The Rational logo, Connexis, ObjecTime, Rational Developer Network, RDN, ScriptAssure, and XDE, among others, are 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.

Portions covered by U.S. Patent Nos. 5,193,180 and 5,335,344 and 5,535,329 and 5,574,898 and 5,649,200 and 5,675,802 and 5,754,760 and 5,835,701 and 6,049,666 and 6,126,329 and 6,167,534 and 6,206,584. Additional U.S. Patents and International Patents pending.

U.S. Government Restricted Rights

Licensee agrees that this software and/or documentation is delivered as "commercial computer software," a "commercial item," or as "restricted computer software," as those terms are defined in DFARS 252.227, DFARS 252.211, FAR 2.101, OR FAR 52.227, (or any successor provisions thereto), whichever is applicable. The use, duplication, and disclosure of the software and/or documentation shall be subject to the terms and conditions set forth in the applicable Rational Software Corporation license agreement as provided in DFARS 227.7202, subsection (c) of FAR 52.227-19, or FAR 52.227-14, (or any successor provisions thereto), whichever is applicable.

Warranty Disclaimer

This document and its associated software may be used as stated in the underlying license agreement. Except as explicitly stated otherwise in such license agreement, and except to the extent prohibited or limited by law from jurisdiction to jurisdiction, 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, non-infringement, title or fitness for a particular purpose or arising from a course of dealing, usage or trade practice, and any warranty against interference with Licensee's quiet enjoyment of the product.

Third Party Notices, Code, Licenses, and Acknowledgements

Portions Copyright ©1992-1999, Summit Software Company. All rights reserved.

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, Visual Basic, the Visual 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, 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.

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

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.

BasicScript is a registered trademark of Summit Software, Inc.

Design Patterns: Elements of Reusable Object-Oriented Software, by Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides. Copyright © 1995 by Addison-Wesley Publishing Company, Inc. All rights reserved.

Copyright ©1997 OpenLink Software, Inc. All rights reserved.

This software and documentation is based in part on BSD Networking Software Release 2, licensed from the Regents of the University of California. We acknowledge the role of the Computer Systems Research Group and the Electrical Engineering and Computer Sciences Department of the University of California at Berkeley and the Other Contributors in its development.

This product includes software developed by Greg Stein <gstein@lyra.org> for use in the mod_dav module for Apache (http://www.webdav.org/mod_dav/).

Additional legal notices are described in the legal_information.html file that is included in your Rational software installation.

Preface

Rational ClearCase is a comprehensive configuration management (CM) system that manages multiple variants of evolving software systems and tracks changes. ClearCase maintains a complete version history of all software development elements, including code, requirements, models, scripts, test assets, and directory structures. It performs audited system builds, enforces site-specific development policies, offers multiple developer workspaces, and provides advanced support for parallel development. ClearCase includes Unified Change Management (UCM), an optional, out-of-the-box process for organizing software development teams and their work products.

About This Manual

This manual guides software developers through everyday development tasks that use either UCM or customizable features of ClearCase.

Organization

The manual is divided into two parts:

- *Working in UCM*. Read this part only if your development team uses UCM to implement its development process.
- *Working in Base ClearCase*. Read this part only if your development team uses customizable ClearCase features to implement its own development process.

Recommended Reading Paths

Use this manual as a guide during your first few weeks of developing software with ClearCase. We recommend this sequence for proceeding:

- Skim through this manual to understand at what points in the development cycle you will need to use ClearCase. Consider skipping the sections titled *Under the Hood*, which describe advanced concepts and technical details.
- Read through this manual and follow the procedures.
- As you become familiar with ClearCase, read the sections titled *Under the Hood*.

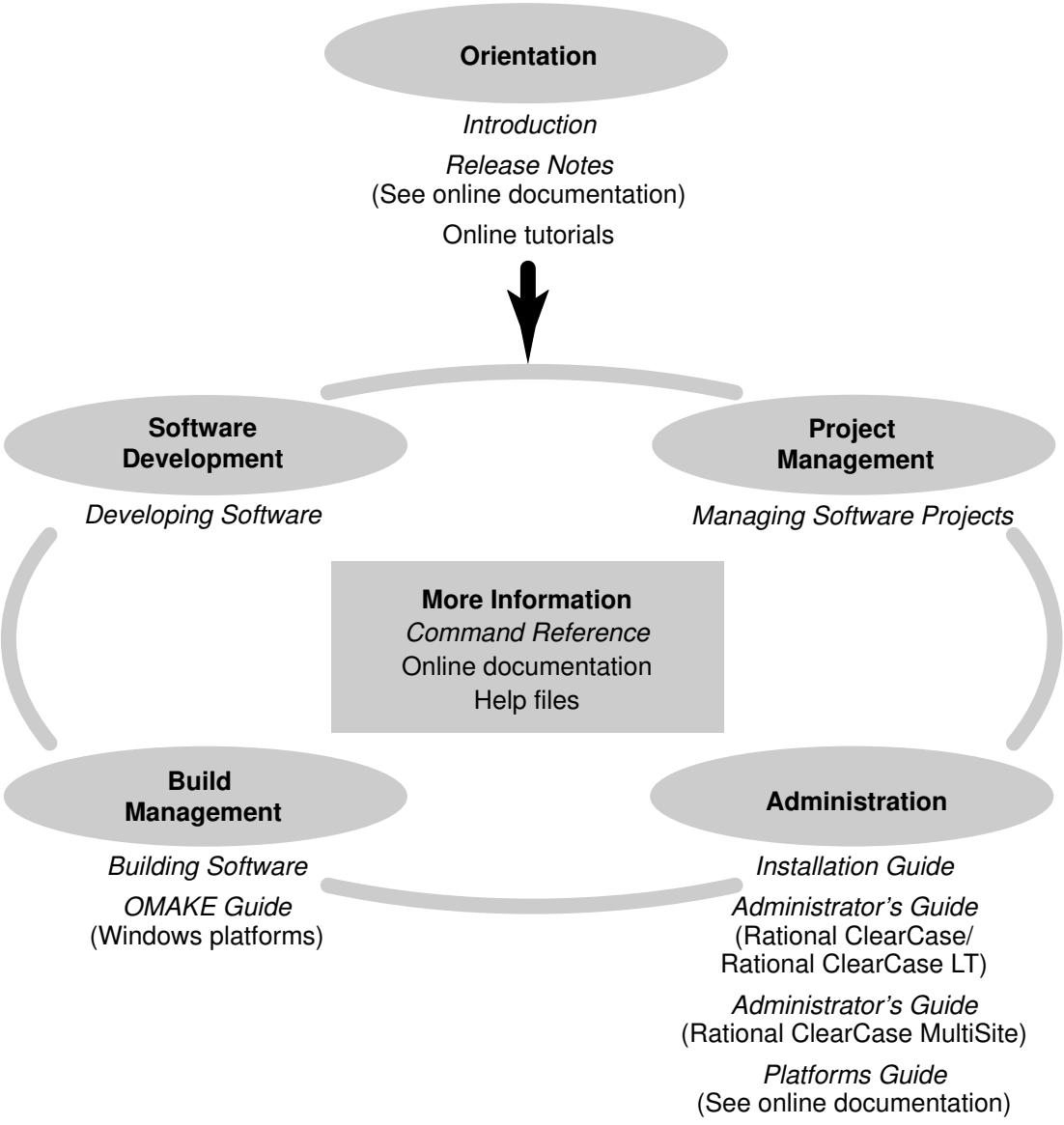
The manual describes using ClearCase primarily from ClearCase Explorer. ClearCase is tightly integrated with ClearCase Explorer; and integrated with Windows Explorer (by means of shortcut menus for ClearCase objects) and with many authoring and development environments. Read this manual to understand the basics of using ClearCase; then see ClearCase Help for information about using ClearCase from your authoring or development environment. For a complete list of ClearCase integrations, see the Rational Web site at **www.rational.com**.

Assumptions

This manual assumes:

- ClearCase has been installed and configured on your computer. For information about installing ClearCase, see the *Installation Guide* for the ClearCase Product Family.
- One or more *versioned object bases* (VOBs), which are the repositories for ClearCase data, exist in your organization.
- If your development team uses ClearCase UCM, your project manager has set up a UCM *project*.
- If your development teams uses the UCM-ClearQuest integration, your project manager has set up an ClearQuest user database that your project uses.
- Your organization has established some other development strategy that uses base ClearCase as the configuration-management system, and you are familiar with this strategy.

ClearCase Documentation Roadmap



ClearCase Integrations with Other Rational Products

Integration	Description	Where it is documented
Base ClearCase-ClearQuest	Associates change requests with versions of ClearCase elements.	ClearCase: *Developing Software* ClearCase: *Managing Software Projects* ClearQuest: *Administrator's Guide*
Base ClearCase-Apex	Allows Apex developers to store files in ClearCase.	*Installing Rational Apex* (UNIX)
Base ClearCase-ClearDDTS	Associates change requests with versions of ClearCase elements.	*ClearCase ClearDDTS Integration*
Base ClearCase-PurifyPlus	Allows developers to invoke ClearCase from PurifyPlus.	PurifyPlus Help
Base ClearCase-RequisitePro	Archives RequisitePro projects in ClearCase.	*RequisitePro User's Guide* RequisitePro Help
Base ClearCase-Rose	Stores Rose models in ClearCase.	Rose Help
Base ClearCase-Rose RealTime	Stores Rose RealTime models in ClearCase.	*Rose RealTime Toolset Guide* *Rose RealTime Guide to Team Development*
Base ClearCase-SoDA	Collects information from ClearCase and presents it in various report formats.	*Using Rational SoDA for Word* *Using Rational SoDA for Frame* SoDA Help
Base ClearCase-XDE	Stores XDE models in ClearCase	XDE Help
UCM-ClearQuest	Links UCM activities to ClearQuest records.	ClearCase: *Developing Software* ClearCase: *Managing Software Projects* ClearQuest: *Administrator's Guide*
UCM-PurifyPlus	Allows developers to invoke ClearCase from PurifyPlus.	PurifyPlus Help

Integration	Description	Where it is documented
UCM-RequisitePro	Allows RequisitePro administrators to create baselines of RequisitePro projects in UCM, and to create RequisitePro projects from baselines.	*RequisitePro User's Guide* RequisitePro Help *Using UCM with Rational Suite*
UCM-Rose	Stores Rose models in ClearCase.	Rose Help *Using UCM with Rational Suite*
UCM-Rose RealTime	Associates activities with revisions.	*Rose RealTime Toolset Guide* *Rose RealTime Guide to Team Development*
UCM-SoDA	Collects information from ClearCase and presents it in various report formats.	*Using Rational SoDA for Word* *Using Rational SoDA for Frame* SoDA Help
UCM-TestManager	Stores test assets in ClearCase.	*Rational TestManager User's Guide* TestManager Help *Using UCM with Rational Suite*
UCM-XDE	Stores XDE models in ClearCase	XDE Help
UCM-XDE Tester	Stores XDE Tester Datastores in ClearCase	XDE Tester Help

Typographical Conventions

This manual uses the following typographical conventions:

- *ccase-home-dir* represents the directory into which the ClearCase Product Family has been installed. By default, this directory is /opt/rational/clearcase on UNIX and C:\Program Files\Rational\ClearCase on Windows.
- *cquest-home-dir* represents the directory into which Rational ClearQuest has been installed. By default, this directory is /opt/rational/clearquest on UNIX and C:\Program Files\Rational\ClearQuest on Windows.
- **Bold** is used for names the user can enter; for example, command names and branch names.
- A sans-serif font is used for file names, directory names, and file extensions.

- **A sans-serif bold font** is used for GUI elements; for example, menu names and names of check boxes.
- *Italic* is used for variables, document titles, glossary terms, and emphasis.
- `A monospaced font` is used for examples. Where user input needs to be distinguished from program output, **bold** is used for user input.
- Nonprinting characters appear as follows: <EOF>, <NL>.
- Key names and key combinations are capitalized and appear as follows: SHIFT, CTRL+G.
- [ ] Brackets enclose optional items in format and syntax descriptions.
- { } Braces enclose a list from which you must choose an item in format and syntax descriptions.
- | A vertical bar separates items in a list of choices.
- ... In a syntax description, an ellipsis indicates you can repeat the preceding item or line one or more times. Otherwise, it can indicate omitted information.

  Note: In certain contexts, you can use "**...**" within a pathname as a wildcard, similar to "*" or "?". For more information, see the **wildcards_ccase** reference page.
- If a command or option name has a short form, a "medial dot" (·) character indicates the shortest legal abbreviation. For example:

  lsc·heckout

Online Documentation

The ClearCase Product Family (CPF) includes online documentation, as follows:

Help System: Use the **Help** menu, the **Help** button, or the F1 key. To display the contents of the online documentation set, do one of the following:

- On UNIX, type **cleartool man contents**
- On Windows, click **Start > Programs > Rational Software > Rational ClearCase > Help**
- On either platform, to display contents for Rational ClearCase MultiSite, type **multitool man contents**
- Use the **Help** button in a dialog box to display information about that dialog box or press F1.

Reference Pages: Use the **cleartool man** and **multitool man** commands. For more information, see the **man** reference page.

Command Syntax: Use the **–help** command option or the **cleartool help** command.

Tutorial: Provides a step-by-step tour of important features of the product. To start the tutorial, do one of the following:

- On UNIX, type **cleartool man tutorial**
- On Windows, click **Start > Programs > Rational Software > Rational ClearCase > ClearCase Tutorial**

PDF Manuals: Navigate to:

- On UNIX, *ccase-home-dir*/doc/books
- On Windows, *ccase-home-dir*\doc\books

Customer Support

If you have any problems with the software or documentation, please contact Rational Customer Support by telephone, fax, or electronic mail as described below. For information regarding support hours, languages spoken, or other support information, click the **Support** link on the Rational Web site at **www.rational.com**.

Your location	**Telephone**	**Facsimile**	**Electronic mail**
North America	800-433-5444 toll free or 408-863-4000 Cupertino, CA	408-863-4194 Cupertino, CA 781-676-2460 Lexington, MA	**support@rational.com**
Europe, Middle East, and Africa	+31-(0)20-4546-200 Netherlands	+31-(0)20-4546-201 Netherlands	**support@europe.rational.com**
Asia Pacific	61-2-9419-0111 Australia	61-2-9419-0123 Australia	**support@apac.rational.com**

Working in UCM

Contents

Preface v
About This Manual v
Organization v
Recommended Reading Paths vi
Assumptions vi
ClearCase Documentation Roadmap vii
ClearCase Integrations with Other Rational Products viii
Typographical Conventions ix
Online Documentation x
Customer Support xi

Working in UCM

UCM Workflows 1
Recommended Reading Paths 1
To Access ClearCase Online Information 1
UCM Projects and Workflows 1
The UCM Cyclic Workflow 2
The UCM Serial Workflow 3
Single-Stream Projects and Rational ClearCase MultiSite 3
UCM Work Steps 3
Setting Up Work Areas 4
Finding and Setting Activities 5
Finding Activities 5
Setting Activities 6
Working on Activities 7
Checking Out and Modifying Source Files 8
Viewing Metadata 8
Checking In and Testing Source Files 9
Indicating Your Progress 9
Delivering Activities 10
Preparing Your Work Areas 10
Starting the Deliver Operation 11
Merging 11

Testing 11
Completing the Deliver Operation 11
Other Considerations for Deliveries 11
Completing Activities 12
Rebasing Your Development Work Area 13
Under the Hood: ClearCase UCM Concepts 13
Elements and Views 14
Projects 14
UCM Activities 15
Components and Baselines 16
Streams and Views 16
Integration Streams 19
Development Streams 20
Feature-Specific Development Streams 21
Your Parent Stream 21
Delivering Your Work 22
Baseline Creation 23
Rebasing Your Work Area 24
About Rational ClearCase Explorer 25
The Explorer Window for ClearCase 25
Shortcut Pane 25
Folder Pane 26
Details Pane 26
Information Pane 26
Toolbars 26
Status Bar 26
ClearCase Explorer Tool Organization 26
Toolbox Tab 27
Shortcut Menu Support 27
Tool Context 28
Site Shortcuts 28
ClearCase Explorer Navigation 28
UCM Project Access 28
Non-UCM View Access 29
View Startup 29
The My Activities Folder 29
URL Access 29
Default Context 30
History of Visited Locations 30
Display of ClearCase Element Information 30

Displaying ClearCase Element Information 30
Initial Use and Appearance of Element Information 31
Ongoing Use and Appearance of Element Information 31
ClearCase Explorer Features for a New User 31
Display of ClearQuest User Database Information 32
Displaying UCM Activities Records 32
Controlling the Result Set Display 33
Controlling Record Content 33
Under the Hood: ClearQuest and UCM Concepts 33
Activities 33
Integrating ClearCase and ClearQuest 34
The UCM-ClearQuest Schema 34
Record Types and Records 35
UCM-Enabled Record Types and UCM Activities 35
Projects Enabled for ClearQuest 35
Queries, Reports, and Charts 37
UCM-Specific Queries 37
MyToDoList 38
UCMCustomQuery1 38
Other Queries 39
State Types and State Transitions 40
State Transitions 40
State Types and State Transitions in a Schema Enabled for UCM 42

Setting Up Work Areas. 43

Starting the Join Project Wizard 44
To Start ClearCase Explorer 44
To Start the Join Project Wizard 44
Choosing a Project 44
Verifying Stream Names 45
Development Stream 45
Integration Stream 45
Choosing a Different Parent Stream 45
Setting Up Your Views 46
Determining the View Types 46
Locations for Snapshot Views 47
Under the Hood: A Snapshot View Storage Directory 47
Locations for Snapshot View Storage Directories 48
To Override the Default Snapshot View Storage Location 49

Drive Mapping for Dynamic Views 49
Path Differences. 49
Under the Hood: View Storage for Dynamic Views 50
Valid Locations for Dynamic View Storage Directories 50
To Choose a Location for a Dynamic View Storage Directory 51
Under the Hood: Activating VOBs for a Dynamic View 52
Choosing Components to Load into Snapshot Views 52
Refining the List of Source Files 52
UNIX VOBs and Case-Sensitivity 53
Loading Elements 54
Accessing Your Development View 54
Accessing Your Development View from ClearCase Explorer 54
Accessing a Snapshot View from Windows Explorer 55
Accessing Someone Else's Snapshot View from Windows Explorer. 56
Accessing Dynamic Views from Windows Explorer 56
Starting Dynamic Views and Activating VOBs 56
To Start a Dynamic View from Windows Explorer 56
Connecting to the UCM-ClearQuest Integration 57
To Determine Whether Your Project is Enabled for ClearQuest 57
Connecting to a Schema Repository Enabled for UCM 57
To Connect to a Schema Repository 58
Logging On to a ClearQuest User Database 58
Setting Up Work Areas in a Single-Stream Project 58

Finding and Setting Activities 61

Finding Activities 62
Finding Activities with the My Activities Folder 62
To Open My Activities 62
Finding Activities with MyToDoList and Other Queries 63
To Start the ClearQuest Client 63
Using MyToDoList 63
To Open the MyToDoList Query 64
Creating Your Own ClearQuest Query. 64
Finding Activities with Project Explorer. 65
Finding Activities with cleartool lsactivity 65
To List the Activity Objects in Your Development Stream 65
Finding Activities with ClearCase Dialog Boxes 66
Creating UCM Activities 66
Creating Activities in a Project Enabled for ClearQuest 67

To Create Activities from the ClearQuest Client 67
To Create Activities from ClearCase Dialog Boxes 68
Creating Activities in a Project Not Enabled for ClearQuest 68
To Create Activities from ClearCase Dialog Boxes 68
To Add Activities from the View Properties Dialog Box 69
Assigning Activities in a Project Enabled for ClearQuest 69
To Make Activities Appear in My Activities and MyToDoList 69
Setting Your View to an Activity 70
Setting Activities from ClearCase Explorer 70
Setting Activities from the ClearQuest User Database 71
Setting Activities with cleartool setactivity 71
To Set an Activity with cleartool setactivity 72
Setting Activities from ClearCase Dialog Boxes 72
Unsetting Your View from an Activity 73
To Unset Activities from ClearCase Explorer 73
To Unset Activities from a Command Prompt 73
To See Which Activity Is Currently Set 73
From ClearCase Explorer 73
From a Command Prompt 73

Working on Activities 75

Checking Out Files 76
To Check Out Files and Set Activities 76
From ClearCase Explorer 76
From the ClearCase Shortcut Menu 77
Checking Out Directories 78
Under the Hood: VOB Links 79
Symbolic Links and Hard Links in Dynamic Views 79
Symbolic Links in Snapshot Views 80
Hard Links in Snapshot Views 81
Caution: Losing Data Because of VOB Hard Links 81
Resolving Checkout Problems 81
Working with Checkouts 83
Viewing Element History 83
To View Element History 83
Comparing Versions of Elements 83
To Compare with a Predecessor 83
To Compare with a Version Other Than the Predecessor 83
Comparing with a Change-Set Predecessor 84
To Compare with a Change-Set Predecessor 84

Tracking Checked-Out Versions 84
Viewing the Change Set of an Activity 85
To View the Change Set of an Activity 85
To View the Change Set of an Activity from Project Explorer 85
Moving Versions to a Different Activity 86
To Move Versions Within a ClearCase Context 86
To Move Versions Within a ClearQuest User Database Context 86
Canceling Checkouts 86
Under the Hood: Canceling Checkouts 86
Canceling Directory Checkouts 87
Canceling the Checkout of a Deleted File 88
Checking In Files 88
To Check In Files 88
Snapshot View: Checking In VOB Links 89
Testing Your Work 89
Indicating Your Progress 89
Modifying Information in a ClearQuest User Database 90
To Modify ClearQuest User Database Information 90
Closing an Activity 90
To Close a ClearQuest Activity 91
Using Schema-Specific Actions 91
Reassigning Activities 92
To Change an Activity Owner 93
Deleting Activities 93
To Delete a UCM Activity 93
To Delete a ClearQuest Activity 94
Working in a Single-Stream Project 95
Tasks While Working in a Single-Stream Project 95
Completing Activities in a Single-Stream Project 96
Finishing Your Activity 97

Delivering Activities 99

Preparing Your Work Areas 100
Finding, Comparing, and Checking In Work from Your Development View . . . 100
To Find, Compare, and Check In Your Work 100
Updating Your Integration View 101
To Update a Snapshot View and Resolve Hijacked Files 101
Starting the Deliver Operation 102
Deliveries and the UCM-ClearQuest Integration 102

Delivering to the Default Target 102
To Start the Deliver Operation to the Default Target 103
Delivering to a Nondefault Target 103
To Start the Deliver Operation to a Nondefault Target 103
Selecting Activities 104
Controlling Merge Behavior 105
Activity Dependencies in the Deliver Operation 105
Baseline Considerations in the Deliver Operation 106
Merging Versions 108
Under the Hood: Concurrent Deliver Operations 109
Under the Hood: Integration Activities and Baselines 110
Using Diff Merge to Resolve Differences 110
To Resolve Differences 111
Under the Hood: How ClearCase Merges Files and Directories 112
Handling a Binary File in a Deliver Operation 112
To Stop and Resume the Deliver Operation 113
Delivering Changes to a Read-Only Component 113
For Intraproject Deliver 113
For Interproject Deliver 113
Testing Your Work 114
Checking Out Versions During Testing 114
Under the Hood: Checking Out Versions from Snapshot Views 115
Undoing a Deliver Operation 115
To Undo a Deliver Operation 115
Completing the Deliver Operation 116
To Complete a Deliver Operation 116
What Happens When You Complete a Deliver Operation 116
MultiSite: Posting Work to Deliver 117
To Deliver to a Nonmastered Target Stream 117
Remote Deliveries with the UCM-ClearQuest Integration 118

Rebasing Your Work Area 119

Preparing Your Development View 120
To Prepare Your Development View 120
Starting the Rebase Operation 121
To Start the Rebase Operation 121
Rebasing to Baselines Other Than Recommended 122
To Rebase to Baselines Other Than Recommended 123
Rebasing Your Development Work Area 123

To Stop and Resume the Rebase Operation 124
Merging Versions 124
Rebasing in a Single-Stream Project 125
Testing Your Development Work Area 125
Checking Out Elements 126
Undoing a Rebase Operation 126
To Undo a Rebase Operation 126
Completing the Rebase Operation 127

Other Development Tasks. 129

Adding Files and Directories to Source Control 129
To Add Files and Directories to an Existing Directory Tree 129
Under the Hood: What Happens When You Add to Source Control 130
To Add Elements for a New Development Task. 132
Importing Files 133
Moving, Removing, and Renaming Elements 133
Moving and Removing Elements 133
To Move an Element Within a VOB 134
Moving an Element to Another VOB 134
To Remove an Element Name from a Directory 134
Other Methods for Removing Elements 135
Renaming Elements 135
To Rename an Element 136
Accessing Elements Not Loaded into a Snapshot View 136
Listing All Elements in the VOB Namespace 136
To See Nonloaded Elements from ClearCase Explorer. 136
To See Metadata for Nonloaded Versions 136
Viewing the Contents of a Nonloaded Version 137
To Copy a Nonloaded Version with cleartool get 137
Adjusting the Scope of a View 137
To Change Which Elements Are Loaded into a Snapshot View 138
Activating or Deactivating VOBs for Dynamic Views 138
To Activate VOBs. 138
To Deactivate VOBs. 138
Assigning a Snapshot View to a Drive 139
Making a Directory Shareable and Assigning a Drive 139
To Enable Sharing on the Snapshot View Root Directory 139
To Assign the Shared Snapshot View Root Directory to a Drive 140
Using the subst Command for Snapshot View Access 140

Moving Work from a Development Stream 141
To Move Work from a Development Stream to Another Project 141
Registering a Snapshot View 142
To Access Another's Snapshot View from ClearCase Explorer 142
To Register a Snapshot View Using Windows Explorer 143
Moving Views 143
Changing the Physical Location of a Snapshot View 144
To Find the Location of the View Storage Directory 144
Update After Moving 144
Moving a View Storage Directory 144
Accessing Views and VOBs Across Platform Types 145
Creating Views Across Platform Types 145
Snapshot View Characteristics and the Operating System 145
Accessing Views Across Platforms of Different Types 146
Accessing UNIX Snapshot Views from Windows Hosts 146
Accessing Windows Snapshot Views from UNIX Hosts 146
Accessing UNIX Dynamic Views from Windows Hosts 146
Accessing Windows Dynamic Views from UNIX Hosts 147
Accessing VOBs Across Platforms of Different Types 147
Developing Software Across Platforms of Different Types 148
Regenerating the view.dat File 148
To Regenerate the view.dat File 148

Working in a Snapshot View Without the Network 149

Setting Up a View for Your Hardware Configuration 150
Under the Hood: View Storage in Disconnected-Use Configurations 151
Preparing the View 152
Disconnecting the View 152
Hardware Configuration: View on a Laptop 152
To Deactivate ClearCase Integrations 152
Hardware Configuration: View on a Removable Storage Device 153
Hardware Configuration: View Copied to a Storage Device 153
Working in the View 153
Hijacking a File 154
To Hijack a File 154
To Find Modified Files While Disconnected 154
Reconnecting to the Network 155
Hardware Configuration: View on a Laptop 155

Hardware Configuration: View on a Removable Storage Device 155
Hardware Configuration: View Copied to a Storage Device 155
Using the Update Tool 155
Determining How to Handle Hijacked Files 155
To Find Hijacked Files 156
To Compare a Hijacked File to the Version in the VOB 156
To Check Out a Hijacked File 156
Merging the Latest Version to a Hijacked File 157
To Undo a Hijack 157
Under the Hood: Determining Whether a File Is Hijacked 158
Other Ways to Handle Hijacked Files 158
Updating the View 158

Index 159

Figures

Figure 1 MyActivities in ClearCase Explorer 6
Figure 2 A View as Seen from ClearCase Explorer 14
Figure 3 UCM Activity Object 15
Figure 4 Elements, Components, and Baselines 17
Figure 5 A Stream Configuration 18
Figure 6 Project Integration Stream 19
Figure 7 Many Development Streams 20
Figure 8 Delivering to the Integration Stream 23
Figure 9 Creating a Baseline for the Integration Stream 24
Figure 10 Rebasing Your Work Area 25
Figure 11 A PVOB Connected to ClearQuest User Databases 36
Figure 12 UCM-Enabled Queries in a ClearQuest User Database 38
Figure 13 Example of ClearQuest User Database State Transitions 41
Figure 14 List of Projects 44
Figure 15 Dynamic Views Started on a Host 50
Figure 16 Clear Check Boxes for Unused Components 53
Figure 17 Development View in ClearCase Explorer 54
Figure 18 My Activities in ClearCase Explorer 62
Figure 19 Linked Activities in MyToDoList Query 64
Figure 20 Activity List in Project Explorer 65
Figure 21 Check Out Dialog Box for a Project Enabled for ClearQuest 66
Figure 22 Select Your View 71
Figure 23 The ClearCase Shortcut Menu from the Open Dialog Box 78
Figure 24 VOB Link 79
Figure 25 Compare with Change Set Predecessor 84
Figure 26 Undelivered Activities 104
Figure 27 Activity Dependencies 106
Figure 28 Baseline Consideration in Deliver 107
Figure 29 Baseline Consideration in Alternative Target Deliver 108
Figure 30 Diff Merge Window 111
Figure 31 ClearCase Merge Algorithm 112
Figure 32 Integration Activity 115
Figure 33 Rebase Stream Preview Dialog Box 121
Figure 34 Development Work Area After Rebasing 124
Figure 35 Creating an Element 131
Figure 36 Directory Structure in a Snapshot View 132

Figure 37 Removing an Element Name from a Directory 134
Figure 38 Renaming an Element 135
Figure 39 View on a Laptop 150
Figure 40 View On a Removable Storage Device 150
Figure 41 Copy the View 151
Figure 42 Hijacked Files in the Results Window 156
Figure 43 Hijacked Version May Not Be the Latest Version 157

UCM Workflows 1

Unified Change Management (UCM) structures the efforts of your software development team into a defined, repeatable process. This chapter describes Rational ClearCase concepts and the UCM workflows and how the flows affect your work as a developer.

Recommended Reading Paths

Read this chapter first. Then, to start working immediately, use Help and the tutorials to learn as you go. Or, if you prefer a more structured approach, use the remainder of *Working in UCM* as a guide through the development cycle.

Until you are familiar with using Rational ClearCase, consider skipping the sections titled *Under the Hood*, which describe advanced concepts and technical details.

Working in UCM refers to tasks performed by both project managers and integrators as well as by developers. For information about tasks performed by other individuals within the UCM workflow, see *Managing Software Projects*.

To Access ClearCase Online Information

To start ClearCase Help, click **Start > Programs > Rational Software > Rational ClearCase > Online Help**.

The help provides links to introductory material. To access the tutorials, click **Start > Programs > Rational Software > Rational ClearCase > Tutorial**.

For more information, see the *Preface* on page v.

UCM Projects and Workflows

When your work team uses UCM, your project manager creates a UCM object called a project. The project is established using one of the following models.

- Multiple-stream—you access source files in a private work area. No one can see your changes. You cannot see others' changes. Then, when you complete your

changes, you make your work available to others in a shared work area. Periodically, you have to update your private work area with the latest changes from the shared work area.

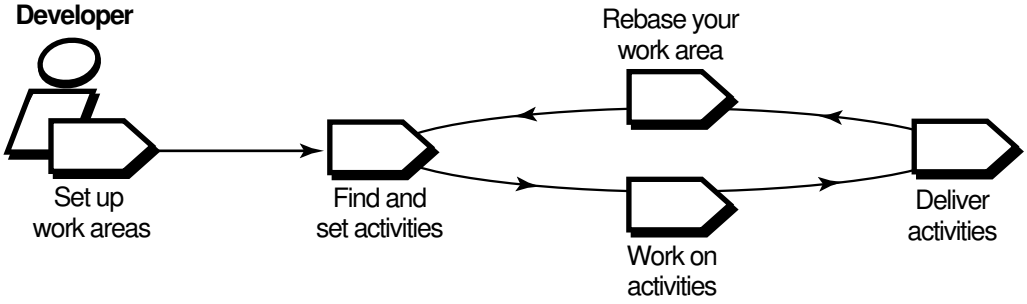
- Single-stream—you access source files from a work area that you share with your team members and work on those files in a private area. You can save your changes in your private area so that other team members cannot see them. However, when your changes are ready to be made available, others can see those changes immediately and can access those files to make their own changes.

In multiple-stream development, a relatively large number of individuals works in isolation in their own streams. Team members control when changes of other team members are visible and are responsible for integrating their changes with those of other team members.

In single-stream development, a small number of individuals works relatively closely in a simplified work environment. In this model, isolation is minimal and integration is fairly rapid because team members do not have to perform an explicit integrate action.

The UCM Cyclic Workflow

In multiple-stream project development, your work as a software or information developer follows a cyclic flow.

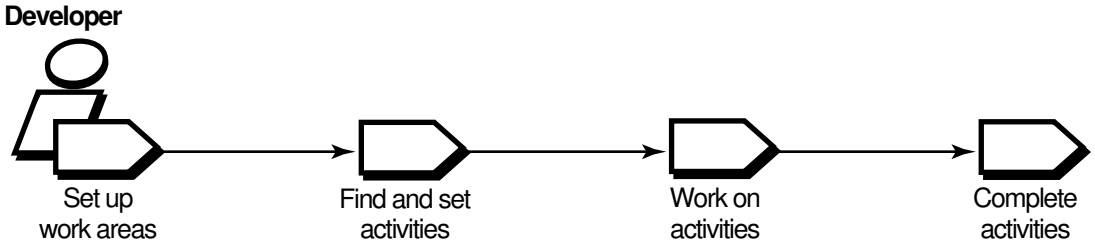


To access your project source files, you set up work areas. As you modify source files, you use activities to organize, identify, and deliver your work.

Regularly, you rebase your work area to take advantage of changes that other members in the project make. You can repeat the steps in this cycle as many times as your project work demands.

The UCM Serial Workflow

In single-stream projects, your work as a software or information developer follows a serial flow.



To access your project source files, you set up work areas. To modify source files, you use activities to organize, identify, and complete your work. When you complete your work, your changes are immediately available to project team members. Unlike in multiple-stream projects, in single-stream projects, there is no delivery or rebasing.

Single-Stream Projects and Rational ClearCase MultiSite

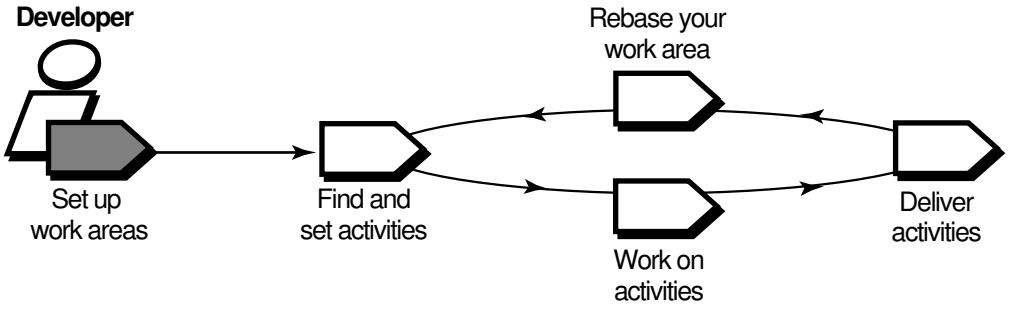
If your organization uses MultiSite, we recommend that you not use UCM serial workflow with teams that are spread across multiple sites and working against multiple replicas. Single-stream projects are not designed for use with teams that are not collocated and working against the same replica. Changing mastership on a single stream restricts people at one site from doing any work while mastership is at another site. Setting up single-stream projects is optimum if the work done in those projects is confined to a single site.

UCM Work Steps

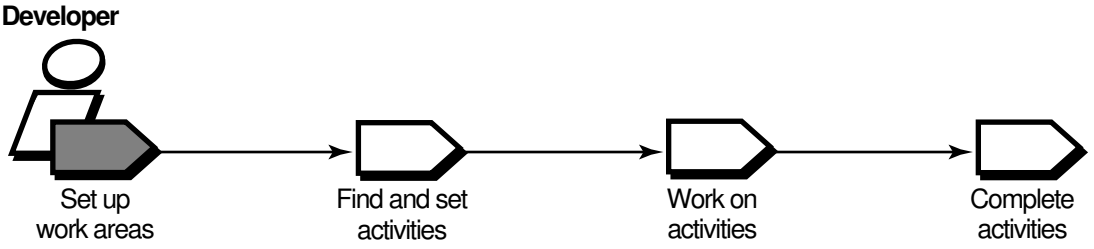
Whether you work in multiple- or single-stream projects, the principles of working in UCM are the same. However, in working in single-stream projects, much of the workflow is simplified in comparison to the workflow in multiple-stream projects.

Setting Up Work Areas

In a Multiple-Stream Project



In a Single-Stream Project



To contribute work to a UCM project, you must set up work areas. This involves joining the project. If you work in a multiple-stream project, you set up two work areas:

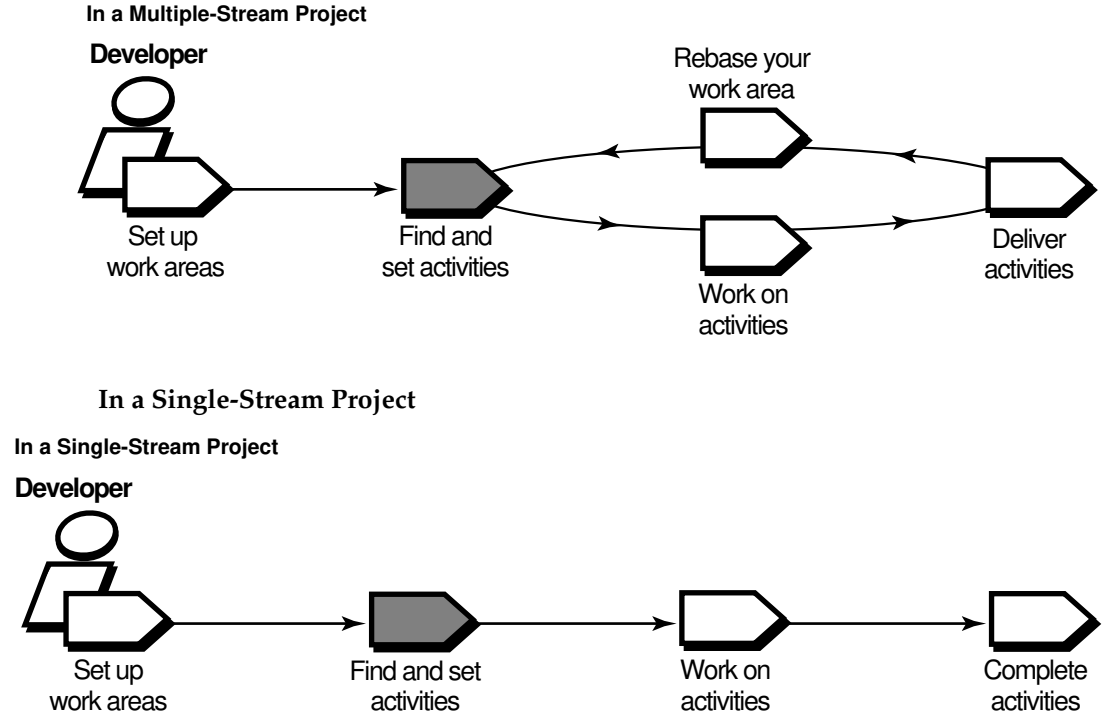
- A private work area for working on activities in isolation
- A shared work area for testing the activities you deliver

A work area consists of a view and a stream. A view is a directory tree that shows a single version of each source file in your project for you to modify. A stream is a ClearCase administrative object that keeps track of activities and baselines and determines which versions of elements appear in your view. For more information, see *Streams and Views* on page 16.

If you work in a single-stream project, you share a stream with other team members and set up your own view in which you work privately. You do your testing in your view.

You can access your views and activities through ClearCase Explorer, your primary interface with ClearCase.

Finding and Setting Activities



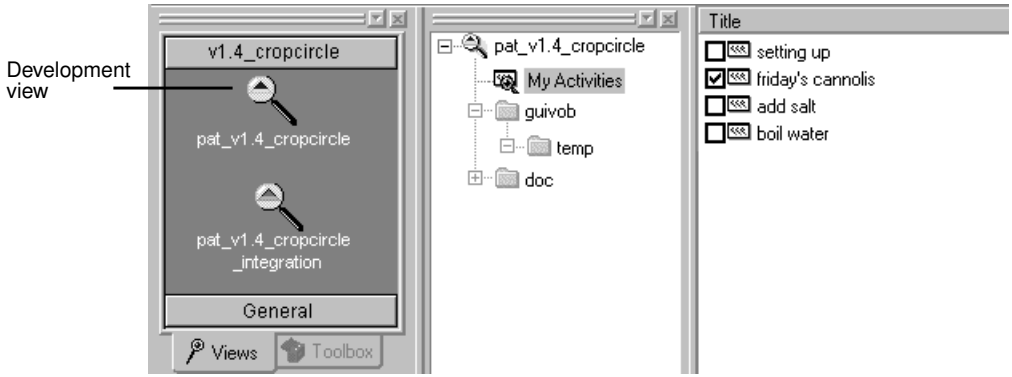
In a Single-Stream Project

After you do your setup, find any activities that your project manager or other team members have assigned to you; if necessary, you can add new ones.

Finding Activities

Clicking the **My Activities** icon in ClearCase Explorer shows the activities that are in your work area (Figure 1). If your project uses the ClearQuest change-request management system, you can use queries such as MyToDoList to find activities that are assigned to you but are not yet in your work area. For more information, see *UCM-Enabled Record Types and UCM Activities* on page 35 and *Queries, Reports, and Charts* on page 37.

Figure 1 MyActivities in ClearCase Explorer

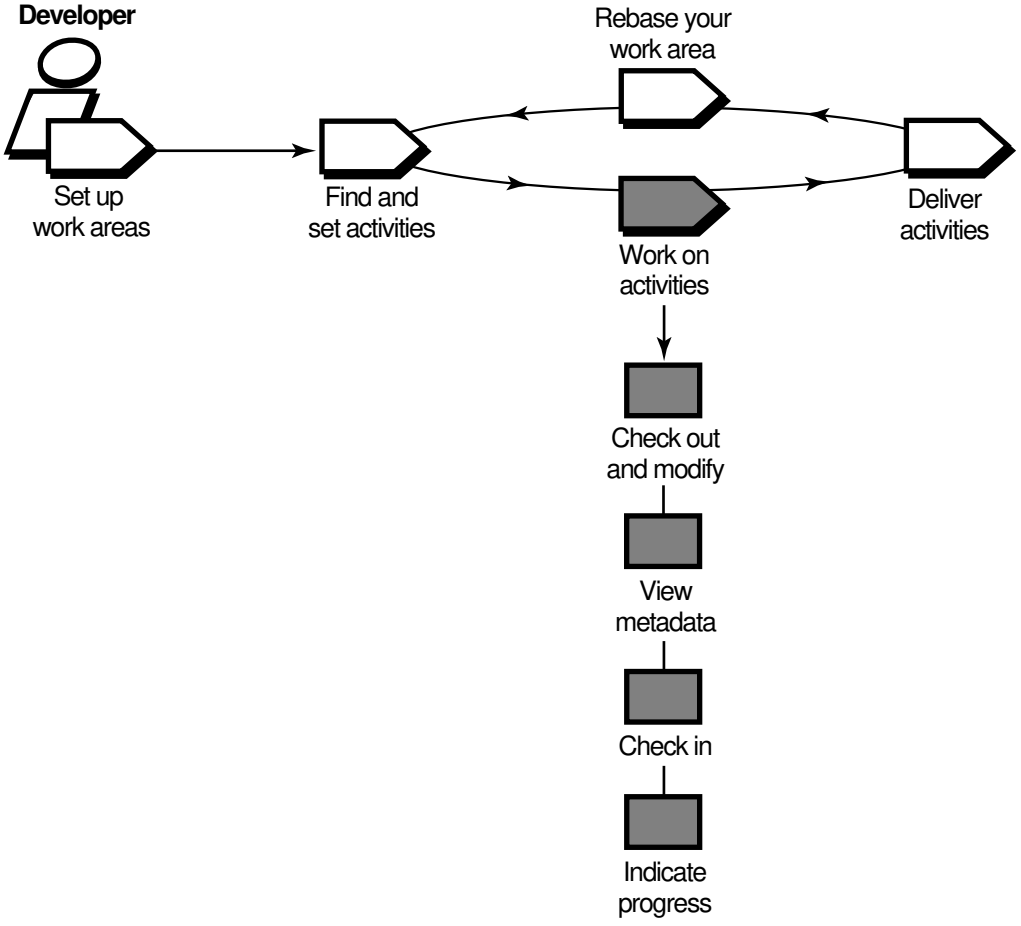


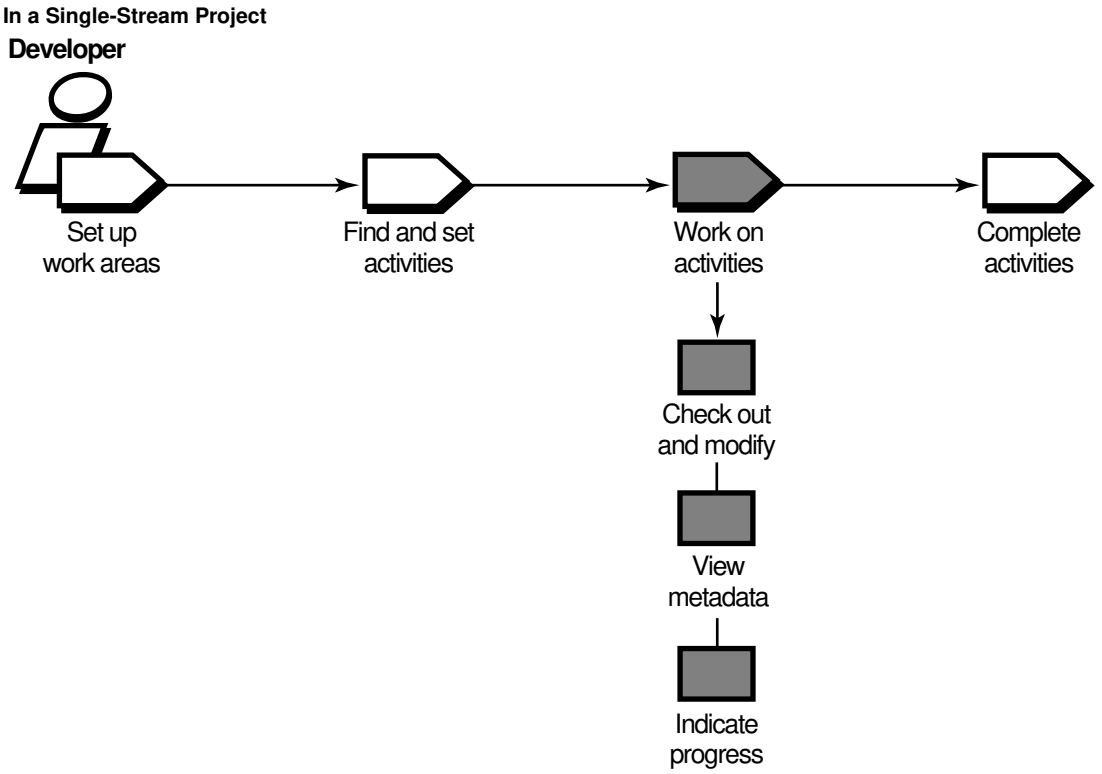
Setting Activities

Setting your view to an activity instructs ClearCase to assign to the activity change set any versions of source files that you create. You must set your view to an activity before you can work on an activity.

Working on Activities

In a Multiple-Stream Project





To work on activities, you check out source files, modify them, and test your modifications. You may also need to enter information in your change-request management system to communicate your progress.

Checking Out and Modifying Source Files

Checking out a file makes it writable in your view. Then you can use any editing tool to modify it.

Viewing Metadata

Metadata is information that ClearCase keeps about your activities and source files. While working with checked-out files (checkouts), you may want to use ClearCase to view the following types of metadata:

- The revision history of an element
- A comparison between versions of an element
- A list of versions that are checked out to your work area

Checking In and Testing Source Files

If you work in a multiple-stream project, when you want to keep a record of the current state of a file, check it in. We recommend checking in to store versions for backup. Any work you check in from your development view is not available to other team members until you deliver it.

Build and test the work in your private work area before it is available to other team members.

If you work in a single-stream project, you do not check in until you test your source files and want to complete your activity.

Indicating Your Progress

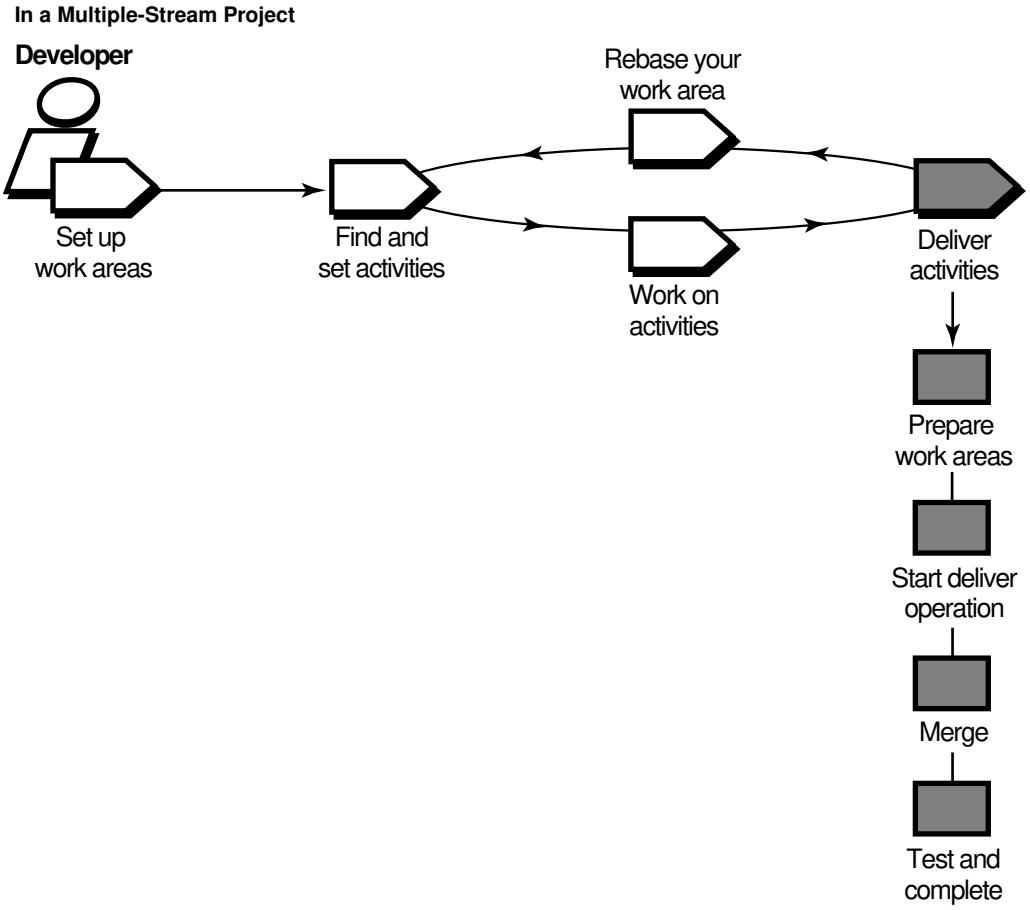
If your project is enabled for ClearQuest, you can indicate your progress on an activity in several ways:

- Modifying information in a ClearQuest user database
- Closing an activity
- Using schema-specific actions

You may need to reassign or delete an activity, regardless of whether your project is enabled for ClearQuest. For more information, see *Indicating Your Progress* on page 89.

On a multiple-stream project, if you are ready to make your work available to the rest of the team, you deliver it (see *Delivering Activities* on page 10). On a single-stream project, if you want to indicate that you are done with your work, you complete your activity (see *Completing Activities* on page 12).

Delivering Activities



If you work in a multiple-stream project, other developers on the project do not see your changes until you deliver your work to a shared work area.

When you are ready to make one or more activities available to the project team, prepare your work areas.

Preparing Your Work Areas

To prepare your work areas:

- If your project integrator has created a new recommended baseline since you last rebased, rebase your development area.

  Periodically, the integrator for your project incorporates activities in the shared work area into baselines, which are sets of activities that represent a significant

change. Then, you synchronize (or rebase) your work area with the activities in the new baseline.

- Find, compare, and check in the work you want to deliver.

Starting the Deliver Operation

After preparing your work areas, start the deliver operation.

Merging

As part of the deliver operation, ClearCase merges the work in your development stream with the work in the shared work area. It completes trivial merges for you. If it encounters merge conflicts, it prompts you to resolve them.

Testing

Your integration view contains the merge results. To make sure that your delivered work is compatible with the work in the shared work area, build and test the files in the integration view.

As part of building and testing, you may need to perform other operations:

- 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 deliver operation.

Completing the Deliver Operation

When you are satisfied with your test builds, complete the delivery. ClearCase checks in your modifications and changes the state of your stream.

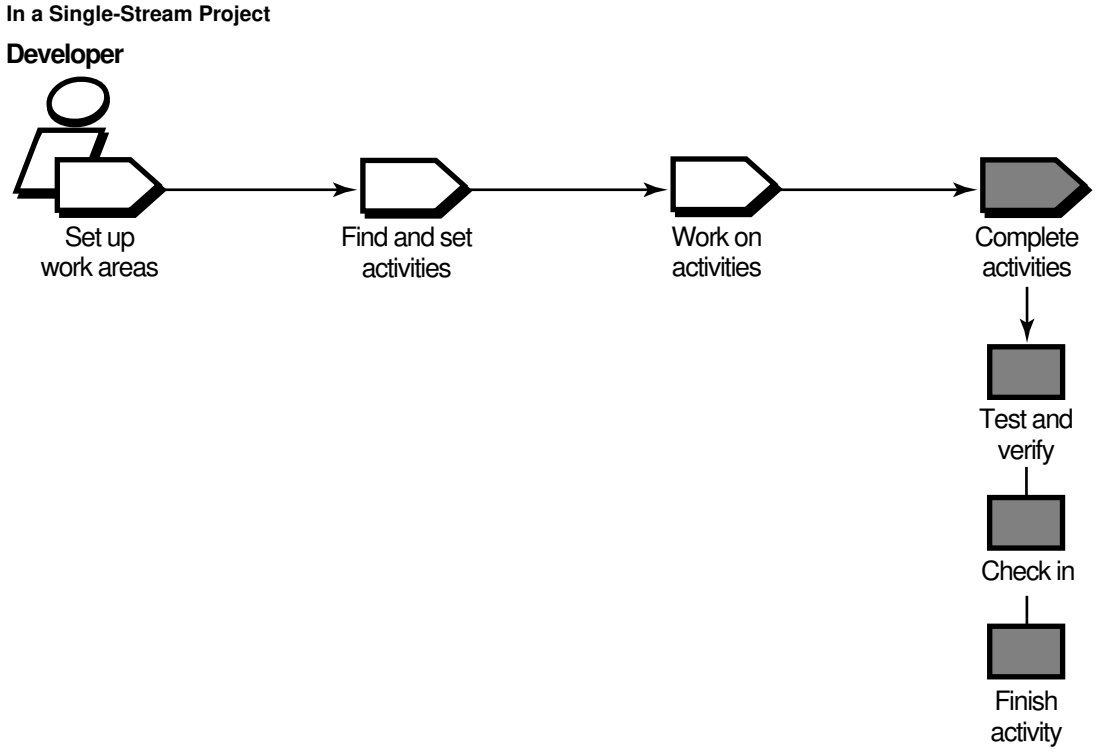
Other Considerations for Deliveries

If your project uses Rational ClearCase MultiSite to share source data with developers in other geographical locations, you may use a different method for delivering activities. If a different site is responsible for controlling your project source data, you do the following:

- Post a deliver operation to the parent stream replica.
- Notify the integrator at the site that controls your project source data.

The project integrator at the other site merges your activities to the parent stream and tests your work.

Completing Activities

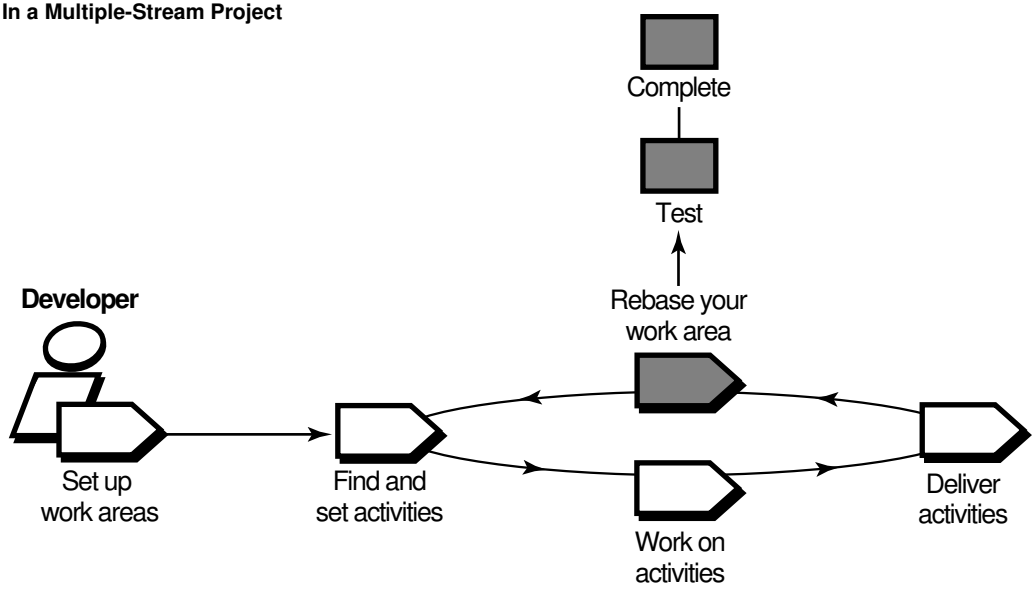


If you work in a single-stream project, other developers on the project see your changes when you check in your work to the shared work area. Because any work you check in from your integration view is immediately available to other team members, you must test your work first.

If your project is enabled for ClearQuest, you indicate that you are done with your work by finishing your activity. For more information, see *Finishing Your Activity* on page 97.

Rebasing Your Development Work Area

In a Multiple-Stream Project



If you work in a multiple-stream project, the project integrator organizes delivered activities into baselines. Usually, baselines go through a cycle of testing and fixing bugs until they reach a satisfactory level of stability. When a baseline reaches this level, your integrator designates it as a *recommended* baseline.

To work with the set of versions in the recommended baseline, you rebase your development work area. To minimize the amount of merging necessary when you deliver activities, rebase your development work area with each new recommended baseline as it becomes available.

After you rebase, be sure to build and test the source files in your development view to verify that your undelivered activities build successfully with the versions in the baseline.

Under the Hood: ClearCase UCM Concepts

This section describes fundamental ClearCase and UCM concepts:

- Elements and views
- Projects and UCM activities
- Components and baselines
- Streams and their relationship to views
- Delivering, creating baselines, and rebasing

Reading this section is not necessary to start working on a UCM project, but it does provide a basis for understanding how ClearCase implements a UCM project and may be helpful if you need to diagnose problems.

Elements and Views

Your project manager places items such as files and directories under ClearCase version control (or source control). Such items are generally categorized as elements. However, files and directories placed under source control are called elements. Each checked-in revision of an *element* is called a *version*.

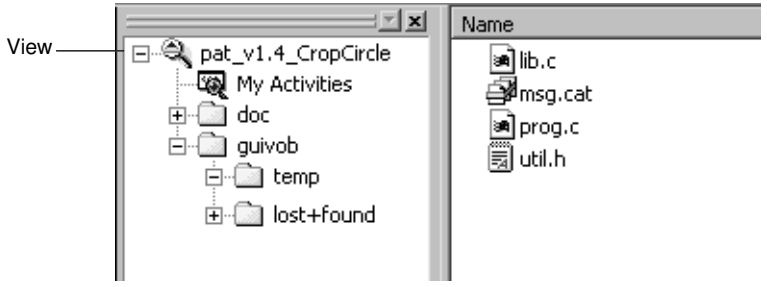
ClearCase stores elements (and their versions) in data repositories called *VOBs* (versioned object bases).

A view provides a directory tree of one version of each file in one or more components. In the view, you modify source files, compile them into object modules for testing purposes, format them into documents, and so on. Figure 2 shows a view named **pat_v1.4_cropcircle** as seen from ClearCase Explorer.

ClearCase offers two kinds of views:

- *Snapshot views*, which copy files from VOBs to your computer.
- *Dynamic views*, which provide immediate, transparent access to the data in VOBs. (Dynamic views are not available on all platforms. For more information, see the Help.)

Figure 2 A View as Seen from ClearCase Explorer



Projects

A *project* is the ClearCase object that defines a set of development policies and a set of configurations used in a significant development effort. Your organization may create a project for each product it develops, for a group of products, for a subset of product functionality, or for a product release.

Policies in a project govern how developers access and modify sets of source files and directories (called components). To record and configure the development work that proceeds on components, projects use other ClearCase objects:

- UCM activities
- Baselines
- An integration stream
- Development streams

A project supports either parallel or serial development. In parallel development, different project members in separate streams can work concurrently with different versions of the same set of source files. In serial development, a small team of project members work from one stream in a simplified work environment.

ClearCase stores projects in a data repository called a *PVOB* (project versioned object base).

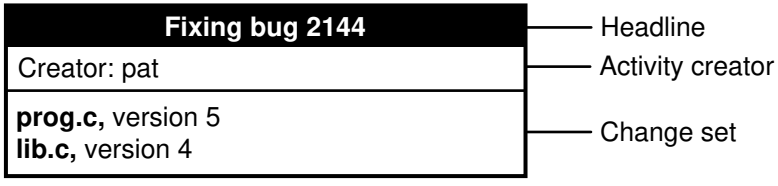
UCM Activities

To create a version in a UCM project, you must first assign it to a UCM activity. A UCM activity is a ClearCase object that identifies the versions created to complete a development task. These versions are referred to as the change set for an activity. For example, the versions you create to fix a defect that is stored in your change request management system may constitute an activity. Your organization determines the scope of its activities.

The UCM activity object includes a text headline, which describes the task, the user ID of the activity creator, and a change set, which identifies all versions that you create or modify while working on the task (Figure 3).

Note: In a project that is enabled for ClearQuest, ClearCase includes a field to describe the activity owner. This activity owner and the UCM activity creator are two different data points; the former is stored in a ClearQuest user database and the latter in a PVOB.

Figure 3 UCM Activity Object



An activity object belongs to one stream and cannot be moved to another stream. (If you assign one or more versions to the wrong activity or if you create new activities to

better represent your work, you can assign the versions to a different activity. For more information, see *Moving Versions to a Different Activity* on page 86.)

When you deliver an activity, ClearCase merges the versions in the activity change set to the target stream, but does not move the activity object to the target stream. Instead, it creates an integration activity to identify the versions created as a result of the merge. For more information, see *Under the Hood: Integration Activities and Baselines* on page 110.

Components and Baselines

The project manager organizes ClearCase elements into components. Usually, a *component* can be built into a functional unit of software. Multiple projects can use the same component. For the project responsible for maintaining the component, the project manager makes the component modifiable. For projects that may use that component to build other components, the project manager makes the same component read-only.

To keep track of different configurations of versions, your project manager or integrator creates a *baseline*, which records one version of each element in a component. In effect, a baseline is a version of a component (Figure 4).

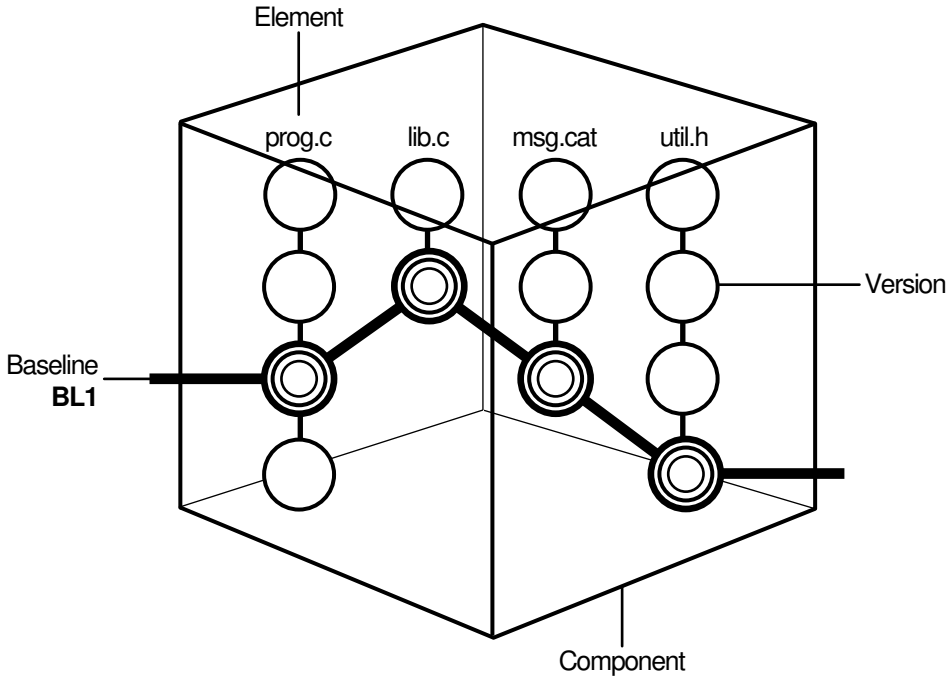
ClearCase stores component and baseline definitions in PVOBs. For more information about components, see *Managing Software Projects*.

Streams and Views

To create new versions of elements in a component, you use a view; to keep track of the versions that you create, you use a stream.

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

Figure 4 Elements, Components, and Baselines

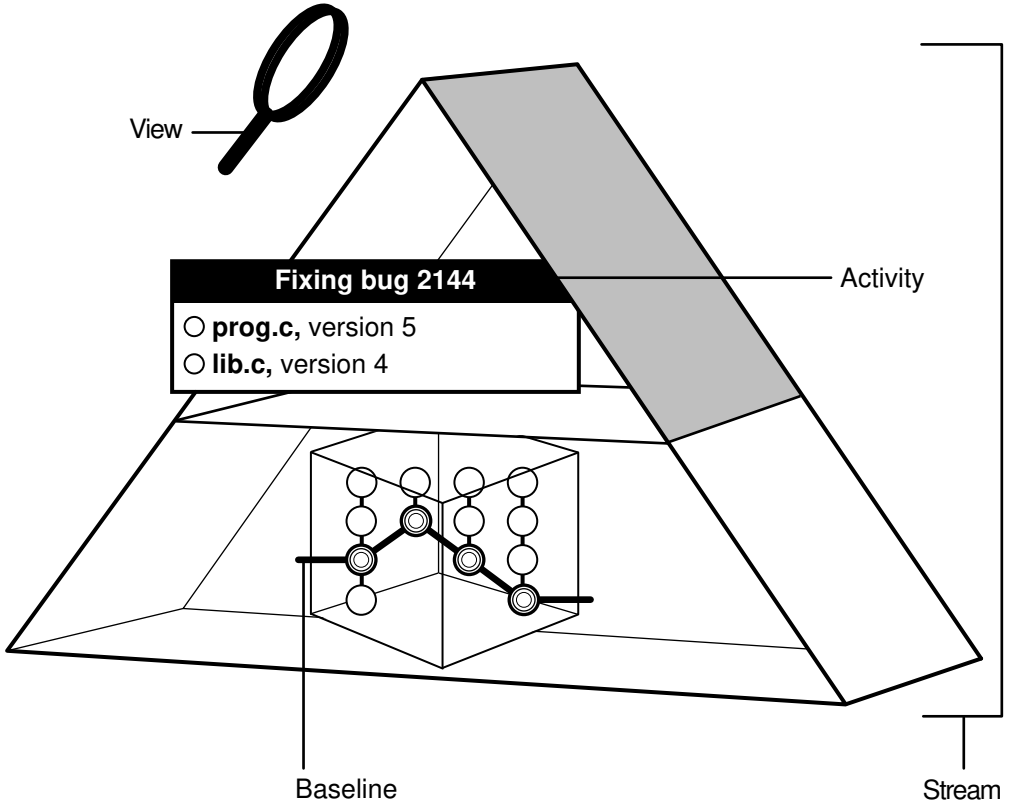


UCM uses baselines and activities to encapsulate a stream configuration in a multiple-stream project. When a stream is created, its original configuration is the same as a baseline (that is, the stream identifies a single version of each element in a component). When you create new versions of elements, you assign the new versions to one or more activities. Hence, a stream configuration is a given baseline plus one or more activities (Figure 5).

Your view accesses versions of elements provided by a stream. The view that you use is said to be attached to the stream. Your view for your private work area is called a development view and the related stream is called a development stream.

A multiple-stream project includes two kinds of streams: an integration stream and development streams. A single-stream project does not allow development streams; your view is attached to the integration stream.

Figure 5 A Stream Configuration



In a project, the following actions modify a stream configuration:

- Checking in versions from a view attached to the stream. (Multiple views may be attached to a stream.)
- Rebasing, which replaces the baseline in the stream configuration with a more recent baseline.
- Delivering activities and baselines (multiple-stream projects only).

  When you deliver activities, you change the configuration of the shared work area. This type of delivery adds activities that were previously available only to the contributing development stream. (Delivering activities does not modify the development stream.)

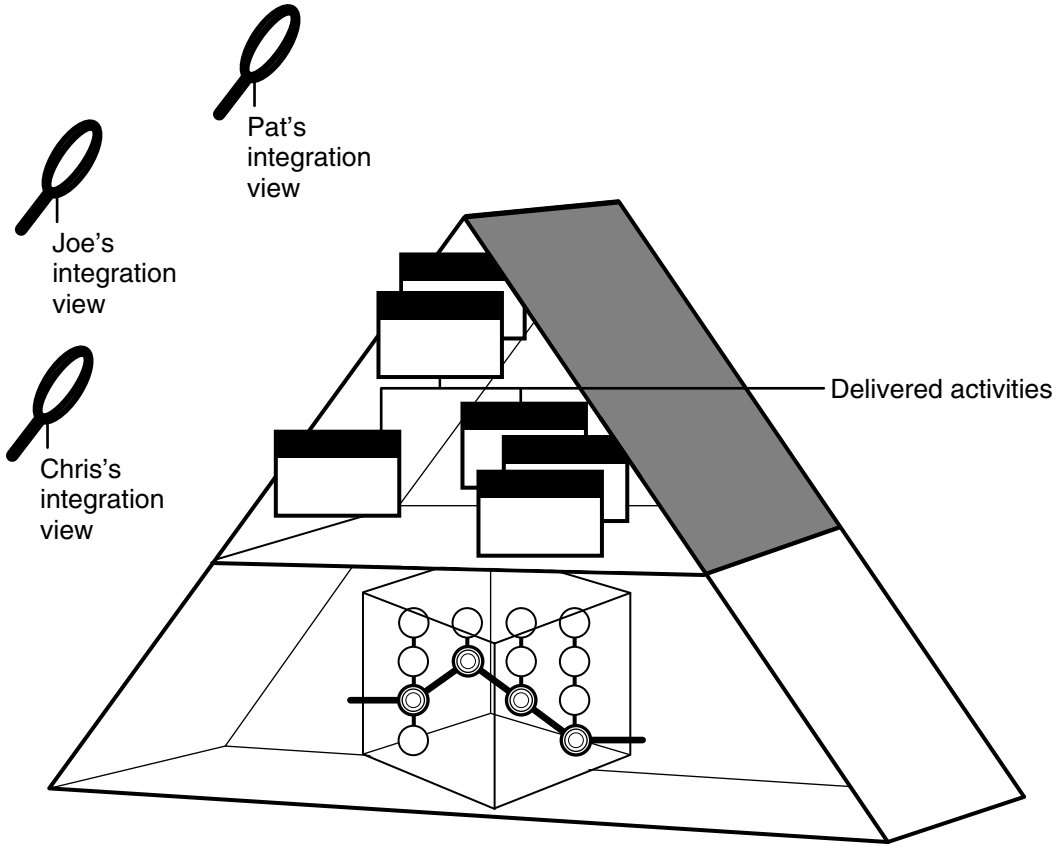
  When you deliver baselines, you introduce new changes to the target stream. This type of delivery is usually performed only by a project integrator.

In a single-stream project, when you check in versions from your view, you modify the configuration of the integration stream. The latest versions in the stream are available to all project members.

Integration Streams

A project contains one integration stream, which is the project main shared work area (Figure 6).

Figure 6 Project Integration Stream



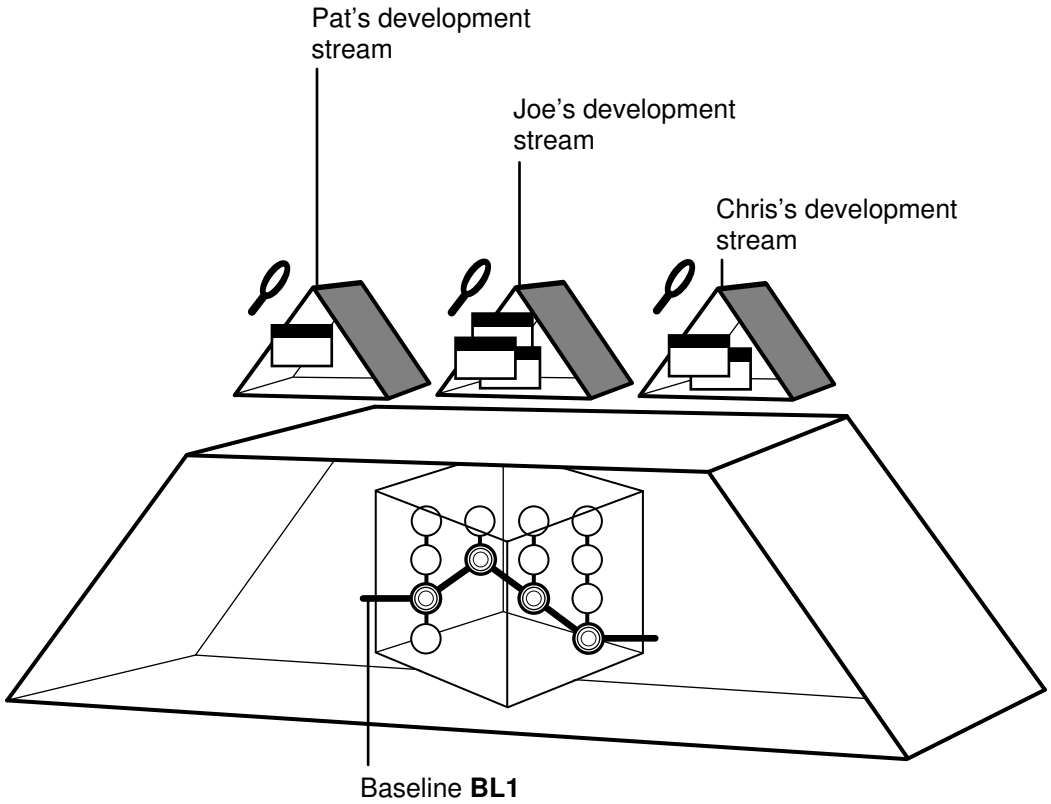
Many integration views can be attached to the project integration stream. The project integration stream records the project baselines and enables access to all versions of the project shared elements. It collects all the work that all team members contribute.

In a single-stream project, all work is done on the integration stream. When you join a single-stream project, you create a view that is attached to the integration stream. The integration stream enables access to the latest versions of the project shared elements.

Development Streams

Typically, each multiple-stream project includes many development streams, one for each developer on the project (Figure 7.)

Figure 7 Many Development Streams



All development streams start from a baseline. As a private work area, your development stream evolves separately from other development streams as you add activities.

Single-stream projects do not use development streams.

Feature-Specific Development Streams

In the basic UCM process, the integration stream is the only shared work area for the entire project. However, a multiple-stream project may have additional shared work areas for developers who work together on specific parts of the project. Using the UCM development stream hierarchy feature, the project can have multiple shared work areas.

A manager of a multiple-stream project can set up a stream hierarchy in which multiple development streams can have child streams. Each parent development stream supports a small team of developers which develops a specific feature. The parent development stream serves as the shared work area for the feature development. For more information about stream hierarchies, see *Managing Software Projects*.

When you join the project, you can create your development stream under a different parent stream than the integration stream, that is, under a feature-specific development stream.

Your Parent Stream

When you join a multiple-stream project, you specify from which stream you will be working. Typically, you join a project at its integration stream. If your project uses feature-specific development streams, you can join the project at the level of the parent development stream rather than at the integration stream.

The stream at which you join the project becomes the parent stream to your development stream and serves as your shared work area. Therefore, if you join the project at its integration stream, the project integration stream becomes your shared work area.

The integration view that you use is attached to your parent stream. Therefore, the integration view shows the baseline for the parent stream and all delivered activities in that stream.

The parent and child relationship defines the default deliver and rebase relationship between the streams. The default deliver relationship says that the child stream delivers to its default target, its parent stream. The default rebase relationship says that the child rebases with the recommended baselines of the parent stream. The default relationships are not modifiable. However, you can deliver your work to a target stream other than the default target.

When you deliver your work, by default, you deliver to the parent stream. Periodically, the project integrator incorporates the delivered work into new baselines. Then, you rebase your development stream to the new recommended baseline of the parent stream.

If you are working from a parent development stream and all the developers finish working on the feature, they deliver their last work to the parent development stream. The integrator then incorporates all the delivered work into a final set of baselines. A parent development stream is an intermediate integration area, the changes in which migrate upward to the project integration stream.

Delivering Your Work

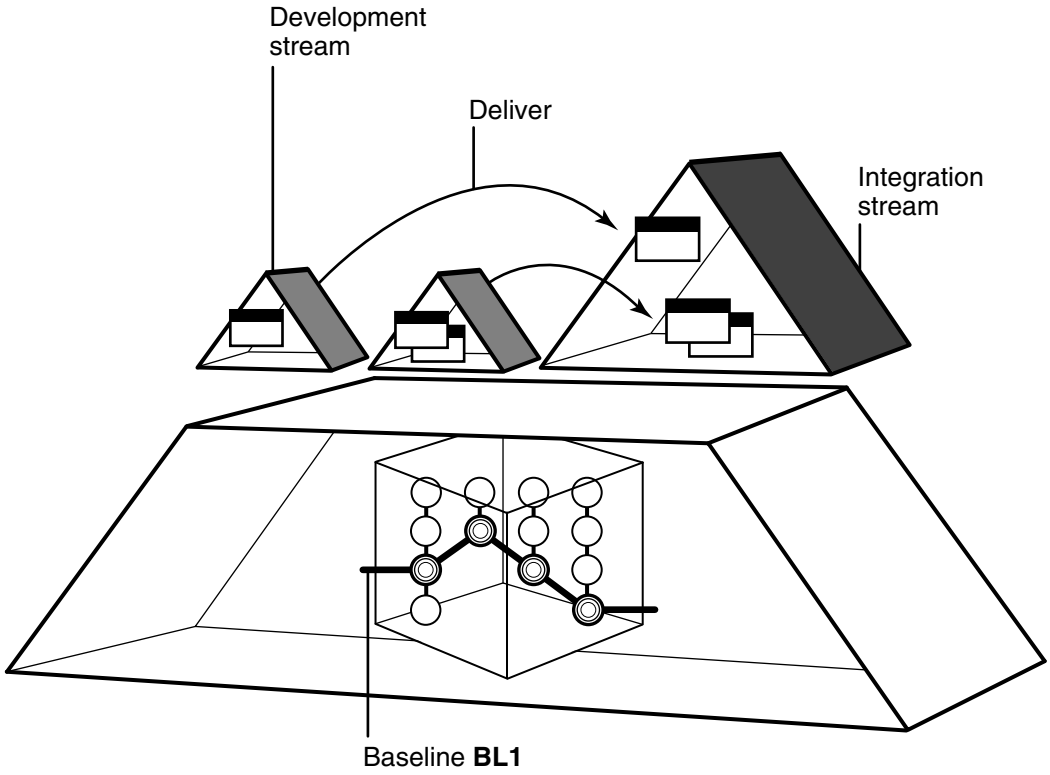
In a multiple-stream project, you start work from a baseline of your parent stream, which becomes your shared work area. To facilitate delivering activities, you have a separate integration view attached to your parent stream. Your integration view shows the baseline for your parent stream and all delivered activities for that stream. By default, you deliver activities from your development stream to your parent stream. The parent stream is the default target for delivering your work.

If your parent stream is the project integration stream, the default target for your deliver operation is the integration stream (Figure 8).

The integration stream identifies a shared set of versions to be used for projectwide building and testing.

If your parent stream is another development stream, the default target for your deliver operation is that development stream (see *Your Parent Stream* on page 21). When you deliver your work, you merge it with the work in your parent development stream rather than with the integration stream. Work from multiple developers is delivered to the parent stream and tested. When all the work in the parent development stream is complete, that work is delivered to its parent stream.

Figure 8 Delivering to the Integration Stream



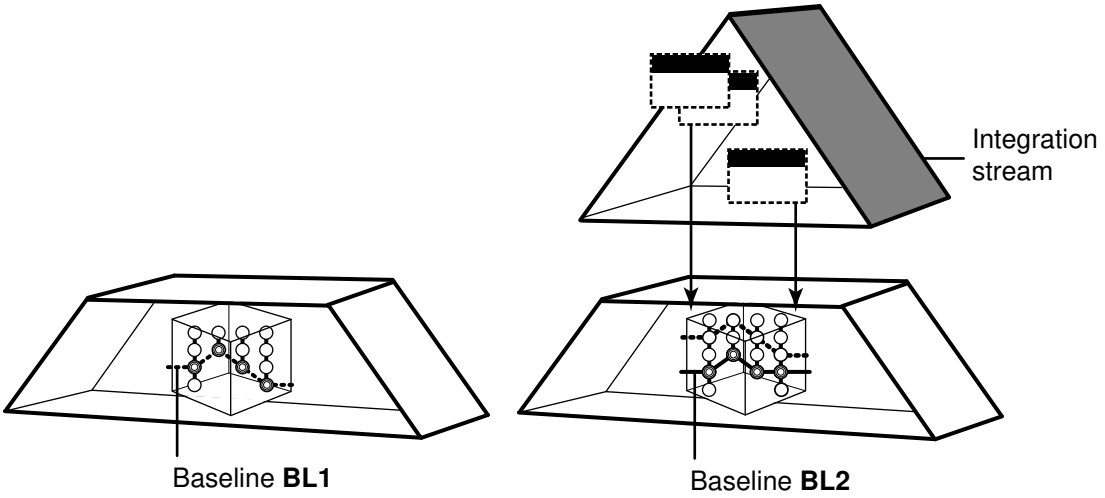
Eventually, the integrators merge all the work from all development streams into the project integration stream.

Baseline Creation

On projects, project integrators incorporate the work delivered to parent streams into new baselines (Figure 9).

For example, the activities with changes delivered to the project integration stream are included in the new baseline **BL2**. The new baseline records the current configuration of the component in the integration stream.

Figure 9 Creating a Baseline for the Integration Stream

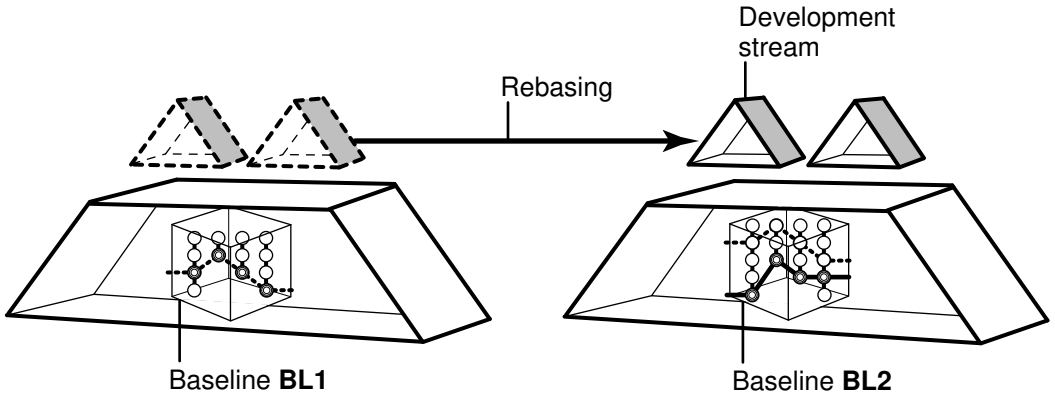


When the integrators build and test these new baselines according to the criteria established for the project, they characterize as recommended the specific baselines that include stable and significant changes. The recommended baseline becomes the new stable configuration for the parent stream. For example, if the integration stream recommends these baselines, they become the required configuration for all development streams whose parent stream is the integration stream. Similarly, for a parent development stream, integrators must recommend new baselines to establish stable configurations for their child development streams.

Rebasing Your Work Area

On a multiple-stream project, you update your development work area with a new baseline by performing a rebase operation. This operation merges files and directories from the parent stream to your development stream and allows you to see the new work identified by the recommended baselines. After rebasing, a development work area shows the versions specified by the baseline, plus any of your undelivered activities (Figure 10).

Figure 10 Rebasing Your Work Area



This allows your work area to stay synchronized with other team member's work. By rebasing often, you minimize the overhead of merging excessive numbers of changes.

About Rational ClearCase Explorer

Rational ClearCase Explorer is a developer's tool that supplies a coherent, adaptable, and customizable interface to your software development environment, providing:

- Organization of ClearCase developer tools
- Access to the ClearCase environment
- Flexible navigation mechanisms

The ClearCase Explorer interface is similar to that of Windows Explorer and has dockable panes. ClearCase Explorer supports shortcuts to ClearCase tools, to ClearCase documentation, and to ClearCase objects.

The Explorer Window for ClearCase

In its initial state, the ClearCase Explorer window has three panes: Shortcut, Folder, and Details (see Figure 1 on page 6).

Shortcut Pane

The leftmost pane, called the Shortcut pane, is dockable and provides tabs to access distinct sections called pages. The **Views** tab accesses icons for shortcuts to ClearCase views and folders in the Folders pane. The **Toolbox** tab accesses icons for shortcuts to ClearCase tools, configurable user tools, and URLs. The icons display ToolTips describing the shortcuts.

Each page organizes icons that provide shortcuts to the items accessible in that page. Click an icon to activate that item. For example, click a view shortcut to start a view. Right-click on the page title to select whether large or small icons are displayed in all pages. If all icons cannot fit on a page, a downward- or upward-facing arrow at the top and bottom of the page provides a scrolling mechanism to access the hidden icons.

Folder Pane

The middle pane, called the Folder pane, is dockable and accesses different tree controls that display the folder and file hierarchy for a given ClearCase view. In the Folder pane, you can navigate to objects in a view that you started.

Details Pane

The rightmost pane, called the Details pane, is not dockable and displays variable content in named tabs having content-specific icons. The content in the Details pane is based on an action in the Folder pane or based on a URL accessed from the **Toolbox** tab.

Information Pane

An optional pane, called the Information pane, is not dockable. When a view is active, the Information pane can provide context-dependent ClearCase state and processing information and access to introductory Help topics.

Toolbars

Similar to the toolbars on your desktop task bar, ClearCase Explorer optionally displays a movable toolbar for elements. The toolbar displays shortcuts to navigation and source control commands and is context sensitive. If you move the toolbar to the desktop, its title bar reads **Standard Toolbar**. You can define your own tailored toolbars.

Status Bar

An optional area in a fixed position at the bottom of the ClearCase Explorer window, the status bar displays information dependent on the current context.

ClearCase Explorer Tool Organization

ClearCase Explorer supports access to tools in several ways:

- Clicking a tab in the Shortcut pane
- Right-clicking in the Folder pane or Details pane
- Clicking an icon in the toolbar (see *Toolbars* on page 26)

Toolbox Tab

The **Toolbox** tab accesses icons for shortcuts to run both ClearCase tools and user-selectable tools and to access documentation and URLs. Tools are organized by tool groups and divided into two pages, **UCM** for project-related tools and **Base ClearCase** for tools not related to UCM projects. The tools provided by default are classified as developer tools. Other ClearCase tools classified as project manager or administrator tools are accessible in the Windows Taskbar by clicking **Start > Programs > Rational Software > Rational ClearCase > Administration**.

The **Toolbox** tab also presents a **Getting Started** page that provides access to Help, online versions of printed documentation, and tutorials.

Shortcut Menu Support

ClearCase Explorer provides access to tools through the following types of shortcut menus:

- For the Windows Explorer integration for ClearCase in the Folder pane and in the Details pane
- For user customizations of the ClearCase shortcut menu
- For customizing the Shortcut pane
- For the UCM-ClearQuest integration

When you right-click in the Folder pane or in the Details pane, a context-sensitive shortcut menu offers access to file handling operations and element processing functions. Provided by the ClearCase-Windows Explorer integration, this shortcut menu contains commands that are activated based on the state of the item you select. If the item is under ClearCase source control, you have available ClearCase commands for element processing functions and Windows commands for file-handling operations. If the item is view-private, you have available file-handling operations and the ClearCase command **Add to Source Control**.

You can use the Context Menu Editor to customize the ClearCase shortcut menu. The editor allows a ClearCase administrator or you to add commands to or modify commands on the ClearCase shortcut menu and add shortcuts to or remove shortcuts from the menu. The steps to do this are described in the Help. You can also add tool shortcuts to your **Send To** folder to access your favorite integrated development environment (IDE), your favorite editor, or a command shell. Any tool shortcuts newly added to your **Send To** folder appear on the shortcut menu **Send To** command the next time ClearCase Explorer starts.

When you right-click in the Shortcut pane, another type of context-sensitive shortcut menu enables you to add and control your own tool groups and tool and URL shortcut icons. If the **Toolbox** tab is active, the shortcut menu commands enable you to add icons

for shortcuts to your favorite tools and URLs and add page controls for your own tools. If the **Views** tab is active, with the shortcut menu, you can add and remove view shortcuts and add pages to organize view shortcut icons. And you can perform some ClearCase operations on views. The ways to do this are described in the Help.

Tool Context

A tool started from the Shortcut pane or from a shortcut menu command can have view context, if applicable. For example, starting the History Browser shows the event log for the element currently selected in the Details pane. If you start a project file (for example, a .DSW or a .VPB file) from within ClearCase Explorer, you can perform element processing functions from within the IDE. As you work with ClearCase objects within the IDE, you can refresh the ClearCase Explorer display to see the state change of the elements in the Details pane.

Site Shortcuts

ClearCase Explorer supports defining tool and URL shortcuts that can be shared among ClearCase users. Called site shortcuts, this type of shareable shortcut is defined externally by a project manager or ClearCase administrator and can be made available to you. (View shortcut definitions can only be locally defined.) The external location for the shortcut definitions is specified during ClearCase site preparation. The location for the definition file is stored in the Windows registry on each ClearCase client computer. You take advantage of these site shortcuts by configuring ClearCase Explorer to import the shortcut definitions at startup. At any time, you can reset system tool shortcuts to remove all modifications or deletions you made to any system or site shortcuts.

ClearCase Explorer Navigation

Through the **Views** tab in the Shortcut pane, ClearCase Explorer provides means for accessing and navigating among ClearCase objects (UCM projects and ClearCase views).

UCM Project Access

Access a UCM project by clicking the **Views** tab in the Shortcut pane. For each UCM project that you join, ClearCase Explorer creates a page with the project name. If you access a UCM project by clicking on its page in the **Views** shortcut pane, the bar for the active page slides to the position immediately above the page contents. In the pane appear the shortcut icons for the development views and the integration view attached to streams in the project (see *View Startup*).

Non-UCM View Access

You access a non-UCM view by clicking the **Views** tab in the Shortcut pane and clicking the **General** page. By default, in the **General** page are icons for shortcuts to non-UCM views that you own.

View Startup

Clicking a view shortcut in a UCM project page or the General page activates (starts) the view, activates the Details pane, and displays the Folder pane. This establishes a *current* view. ClearCase Explorer displays in the Folder pane a hierarchy rooted at the view tag, and, as subfolders under the view, the contents of the view. For a dynamic view, the subfolders are for components of UCM projects and for all mounted VOBs. For a snapshot view, the subfolders are for all VOBs rooted at the view. By default, in a snapshot view, only loaded elements are visible. You can change this default so that unloaded elements are displayed. The Details pane is activated and lists the subfolders that are exposed in the Folder pane.

The title bar shows the currently selected path within the view. The status bar shows the view tag associated with the current view, the name of the current UCM activity set in the view, the number of items listed in the Details pane, and the number of listed items that are selected.

The My Activities Folder

One of the folders under a view for a UCM project is labeled **My Activities**. The contents depends on whether you are using the UCM-ClearQuest integration. In a project that is enabled for ClearQuest, click this folder to see listed in the Details pane all change requests assigned to you. A plus sign (+) designates all unresolved change requests assigned to you and not bound to a UCM activity. Every view shows the same list of unassigned change requests. When a change request is bound to a UCM activity, it is then listed only under the associated activity.

In a project that is not enabled for ClearQuest, the **My Activities** folder lists in the Details pane all UCM activities that are in your work area. In the list of activities, the current activity is indicated by a checked box.

URL Access

You access a URL by clicking the **Toolbox** tab in the Shortcut pane, the page on which the shortcut is located, and the shortcut to the URL. An Internet Explorer control is activated in the Details pane and the target location is accessed. The Details pane renders the HTML content of the target. The URL tab on the Details pane shows the path to the URL. The Folder pane remains unchanged, but the Information pane, if

displayed, is occluded. You can navigate HTML pages clicking **Go > Back** and **Go > Forward**. The status bar maintains the context for the previously selected view.

Default Context

Clicking an icon in the Shortcut pane establishes that shortcut as the default of its type: view. ClearCase Explorer saves this default context when the tool exits, so that it is restored when the tool restarts.

Like the Details pane, the Folder pane maintains navigation context for the current view. Clicking a URL shortcut does not disturb the navigation context. This context is maintained across ClearCase Explorer starts.

History of Visited Locations

As you navigate from one location to another, ClearCase Explorer stores the path for each of your visits. To return to a previously visited location, click **File** and, from the numbered list of paths (ordered from most recently visited to first visited), select the path.

Display of ClearCase Element Information

The tree controls in the Folder pane provide navigation to ClearCase objects in a view (see *ClearCase Explorer Navigation* on page 28).

Displaying ClearCase Element Information

If you activate the Folder pane, the default view is represented as a top-level folder in the Folder pane with the context of the last navigation. Expanding the subfolders provides details about the ClearCase objects accessible in the view. Selecting a folder displays fields of file system and ClearCase information in the Details pane. Use the controls in the Folder pane to expand and close the subfolders and navigate to the element that you want to work on. In the Details pane, elements possessing unique characteristics (like a file under version control or an eclipsed version) display with visual characteristics familiar in other ClearCase tools.

You can control which fields are displayed in the Details pane. Because gathering some data for display takes more processing time and affects tool performance, you may not want to display all possible fields. You can also use standard Windows techniques to sort the data in the columns. Any customizations you make are persistent across invocations of the tool.

In the Folder pane and Details pane, normal Windows Explorer functions such as delete, cut and paste, and drag and drop are supported. Delete (for example, pressing

the DELETE key) performs a **cleartool rmname** operation on a selection. Rename does a **cleartool mv** operation. Cut and paste and drag and drop operations enable you to populate your view with folders and files from the file system.

Initial Use and Appearance of Element Information

If you start ClearCase Explorer on a computer on whicth you have not joined any projects or created any views, the Shortcut pane on the **Views** tab contains only the **General** page. If you join a project, ClearCase Explorer creates a page under the **Views** tab with the project name.

If you create a view associated with a project from within ClearCase Explorer, ClearCase Explorer creates a shortcut for that view on the page for that project. If you are running ClearCase Explorer and, from ClearCase Explorer, start a view that is associated with a project that does not have a page, a bar is created for that page.

For any view that you create from within ClearCase Explorer that is not associated with a UCM project, ClearCase Explorer adds to the **General** page an icon with the view tag. (There is no way to tell ClearCase Explorer not to create an icon for a view that you own.) When ClearCase Explorer exits, it stores, as the default view, information about the view that you last worked in. When ClearCase Explorer starts again, it restores the default view.

Ongoing Use and Appearance of Element Information

Use the **View** menu to adjust the window appearance. Click the check boxes to toggle the display of window panes and bars. Click the **View > Refresh** command to have ClearCase Explorer examine the system for the latest state information and redisplay the information in the Folder pane and Details pane.

If you remove a view that you own, its shortcut in ClearCase Explorer is removed. If you remove all the views for a UCM project, its page in ClearCase Explorer is removed. To remove a shortcut for a view that you do not own, remove its view tag from the registry.

ClearCase Explorer Features for a New User

For new users, ClearCase Explorer offers an Information pane whose contents depend on the context of your work in a view. The Information pane presents workflow help topics that outline the stages required while working in different ClearCase environments.

As you work with ClearCase objects in a view, the Information pane displays contextual information about a selected element. Included is brief information about a selected element and its current state. For example, if you select a file, the Information

pane shows two columns, one listing the details of your selection and the other offering, under the heading **About**, informative text or, under the heading **Uses**, tips on some of the operations that you can perform with the selected object.

In a view context in ClearCase Explorer, you can work in the Details pane where you can create a new file or directory. If you select the newly created file or directory, the Information pane tells you that you can add it to source control. If you add the file or directory to source control, the Information pane keeps the item selected and tells you that you can check it out. If you are working within a UCM project, you can select an activity; or, if you are checking a file out, you can create a new activity. If the project is enabled for ClearQuest, you can select a change request.

After you learn the environment, you can hide the Information pane to reclaim the screen area.

Display of ClearQuest User Database Information

Rational ClearQuest provides a user database of change-request management information that can integrate with UCM projects. Access to a ClearQuest user database is through the ClearQuest client. The tree controls under Workspace in the ClearQuest client provide navigation to folders and queries, charts, and reports in a ClearQuest user database (see *Connecting to the UCM-ClearQuest Integration* on page 57).

Displaying UCM Activities Records

Under Workspace in the ClearQuest client, you see two high-level folders, **Personal Queries** and **Public Queries** (see *Queries, Reports, and Charts* on page 37). The **Public Queries** folder typically contains database search requests provided with the software and by your project administrator.

When you run a query (see *Queries, Reports, and Charts* on page 37), the result set displays data from activity records returned from the ClearQuest user database, one line per record. Filters defined in the query limit the number of records returned. If you run a query without any filters, all of the records in the database are returned.

The Record form, displayed below the Query builder pane, shows data and state information for the activity record whose line is selected in the result set. Clicking another line updates the Record form with information for the newly selected record. Access different fields and state information by clicking tabs in the Record form. The tabs available vary based on integration support. When you click a tab whose area is not visible, the software accesses and displays the related information from the database.

Typically, entries in **Public Queries** are read-only. However, you can modify a public query for your personal use. For example, you could run a public query that found all open defects for all components. Then, you could click **Query editor** in activity details to modify the filters to find such records for just your components. You can run a modified query to sample its results. When you are satisfied, you can save it under a folder in **Personal Queries**. You can also export queries to a file to share them with other users.

Controlling the Result Set Display

Click **Display editor** to control which fields are displayed in each line for records returned in queries. Double-click a column heading to change sort order in the column. You can also right-click to use commands on a shortcut menu.

Controlling Record Content

Buttons to the right of the record form provide commands used to modify fields and transition records. Commands available on these buttons vary with the integrations distributed with ClearCase. A control in the bottom left corner of the record form allows you to single-step through records in a result set, jump to the first or last record, and provide an ID of any record to retrieve.

Under the Hood: ClearQuest and UCM Concepts

The fundamental concepts for the UCM-ClearQuest integration are:

- Activities
- Integrating elements and activities
- Schema enabled for UCM
- UCM-enabled record types and UCM activities
- Queries, reports, charts, and UCM-specific queries
- State types and transitions

Reading this section is not necessary to start working on a UCM project that is enabled for ClearQuest, but it does provide a basis for understanding how ClearCase implements the integration and may be helpful if you need to diagnose problems.

Activities

Your UCM project manager can use ClearQuest customizable defect and change tracking features to set up a repository to store information about your work. Information used to define, schedule, and track your work is referred to as an *activity*. Using these features, your project manager can manage every type of change activity

associated with software development, including enhancement requests, defect reports, and documentation modifications. A repository that stores this information is referred to as a ClearQuest user database.

Integrating ClearCase and ClearQuest

Your project manager can customize your work environment to fit activity management into your UCM workflow. Projects can be enabled for the UCM-ClearQuest integration. Your project manager can relate the elements and versions on which you work to the information about your work. For example, a ClearQuest user database could be set up to contain defect reports about software for which you have development responsibility. When you start working on the versions that relate to the defect, the software asks you to provide information that is later stored with the defect. Your project manager can then examine information in the ClearQuest user database to see the state of your work regarding that defect. The same capability can be used to track progress on requests for enhancement on the software that you work on.

If your project manager integrates elements and activities, a schema that is enabled for UCM is provided. Your project manager can tailor the schema to the ways that your organization does its work.

The UCM-ClearQuest Schema

ClearQuest includes predefined schemas that provide change and defect tracking processes and integration with UCM projects.

A *schema* defines the types of records in a ClearQuest user database, the states available to each type of record, and other attributes of the database. A schema basically defines the process for how developers work with records in a ClearQuest user database. An organization may have many different ClearQuest user databases, each one using different schemas for varying types of records tailored for specific purposes.

A schema includes:

- Record type definitions
- Forms used to submit and modify a record
- Field definitions and behavior
- States a record can be in
- Actions used to modify or change the state of a record
- Hooks for scripting languages that allow fields and actions to be customized

All schemas are stored in a schema repository, which is a master database of schemas. The schema repository does not contain any user-owned activity data or change request data. Change-request data is stored only in a ClearQuest user database.

ClearCase supplies two schemas enabled for UCM and a ClearQuest user database: Unified Change Management and Enterprise. Your project manager can set up a custom schema enabled for UCM. To support the UCM-ClearQuest integration, a ClearQuest user database must use one of these schemas enabled for UCM.

Record Types and Records

Each change request is stored as a record in a ClearQuest user database. A record type is template that defines the actions, fields, behaviors, and possibly state information associated with a record. Schemas can contain state record types for records that can move from state to state and stateless record types for records that remain static.

An activity record has associated states and actions. The state indicates what stage in processing a record is in. A developer uses an action to change the state of a record. The schema also governs the transition from one state to another, determines which actions can change which states, and which states are possible from other states.

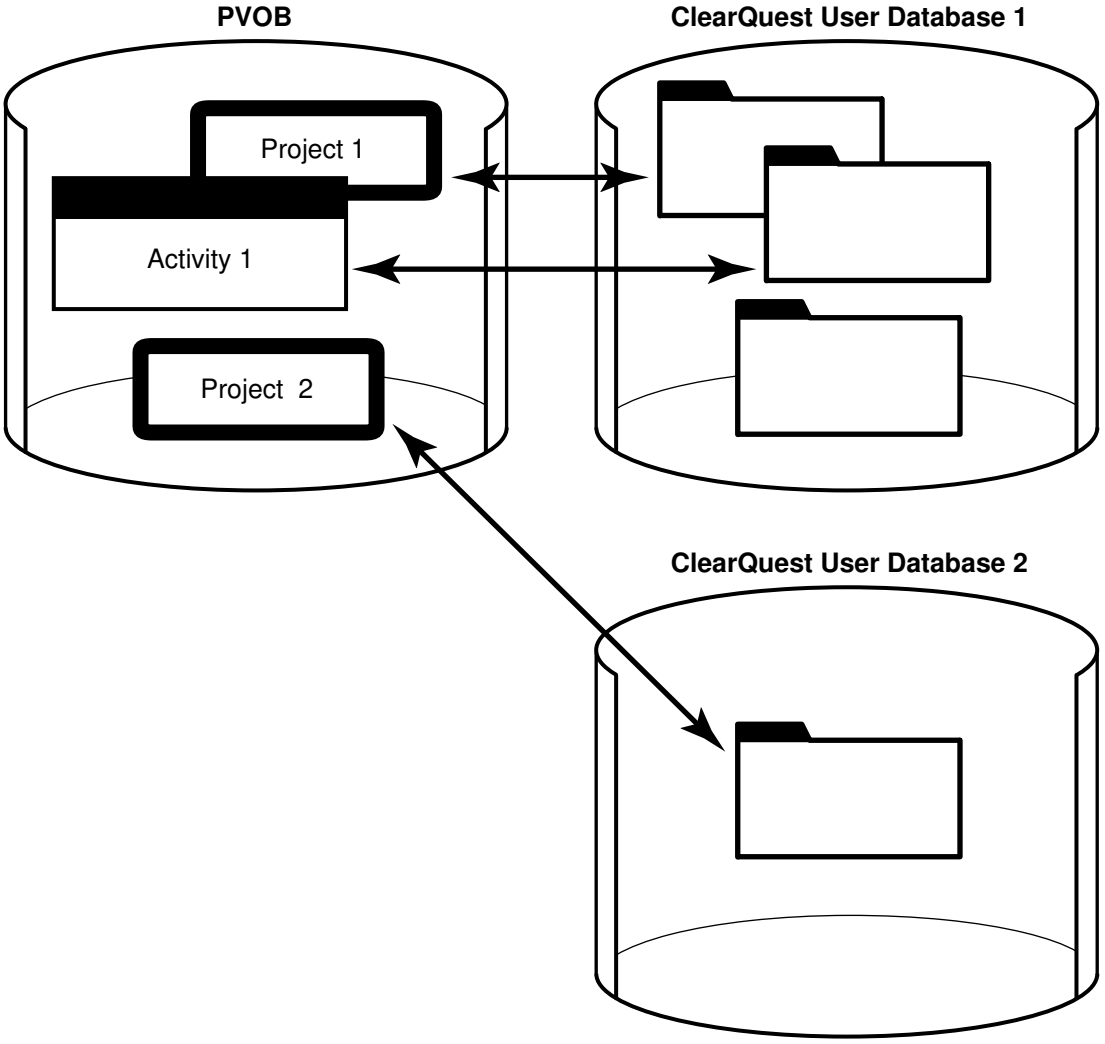
UCM-Enabled Record Types and UCM Activities

A UCM-enabled record type is a template that includes definitions for a set of fields that stores information about UCM activities. Your ClearQuest user database may include different UCM-enabled record types for different purposes (such as tracking defects and enhancement requests), and it may include other record types that are not UCM-enabled.

Projects Enabled for ClearQuest

If a project is enabled for ClearQuest, ClearCase provides links between PVOBs and ClearQuest user databases (Figure 11). A ClearQuest user database stores change requests as user-owned records.

Figure 11 A PVOB Connected to ClearQuest User Databases



One link connects the project in the PVOB with a ClearQuest user database. Multiple ClearQuest user databases can be used. Another link connects the project UCM activity objects in the PVOB to UCM-enabled records in the project associated ClearQuest user database.

UCM activity objects in the PVOB can be linked with activity records based on a UCM-enabled record type in the ClearQuest user database (see Figure 11). The link enables ClearCase to display information about the UCM activity (such as its change set, its stream, and whether it is currently set in any view). It also enables policies that govern when you can deliver an activity described in the PVOB and when you can

close an activity in the ClearQuest user database. Because of the close association between linked UCM activities and UCM-enabled records, the UCM documentation usually refers to both linked entities as *activities*.

At any point in the cycle of a project, your ClearQuest user database may contain UCM-enabled records that are not linked to a UCM activity object. For example, a newly created record may not be linked to a UCM activity. One way to link a UCM-enabled record to a UCM activity is to explicitly complete an action on the record (for example, by clicking **Action > WorkOn** in a ClearQuest user database).

Each UCM activity in a UCM project that is enabled for ClearQuest must be linked to an activity record. You cannot create a UCM activity without linking it to a UCM-enabled record in a ClearQuest user database.

Queries, Reports, and Charts

Queries are the means for navigating through the ClearQuest user database. Before viewing or modifying records in a ClearQuest user database, you must query the database to find the records you are interested in. For example, you may want to see only activities that are assigned to you or that are associated with your project, or you may want to see activities created before or after a particular date.

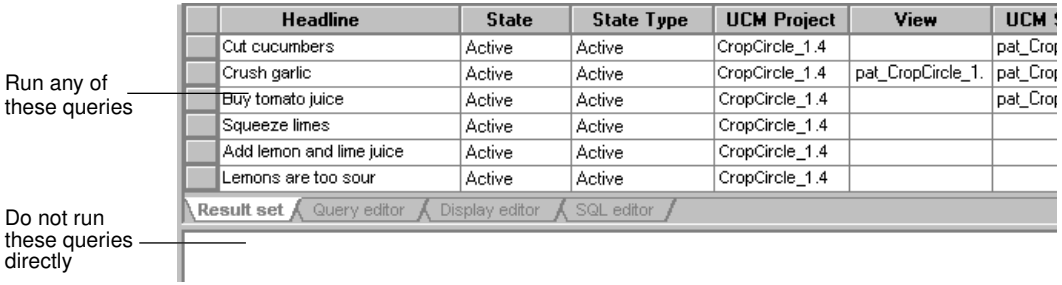
Reports compile information from a ClearQuest user database and generate the information in human-readable format. Associated with a report is a query and a report format. Typically, a project manager creates a report format and saves it in the ClearQuest user database for use with queries. To use a report, you select a report format and a query to "plug in" to the report format. Accessed under Workspace in the ClearQuest client, reports can be run at any time.

Charts are the means to graphically view and analyze data from a ClearQuest user database. Typically, a project manager uses the chart wizard to define one of the following types of chart: distribution, trend, or aging. You can filter a chart with the Query editor. You can display a chart in either pie, bar, stacked bar, area, or line format. You cannot edit query records returned while you are creating a chart.

UCM-Specific Queries

A schema enabled for UCM includes a set of queries that you can use to find records enabled for UCM. ClearCase places the queries into two categories in the Public Queries folder (Figure 12).

Figure 12 UCM-Enabled Queries in a ClearQuest User Database



You can run queries in the UCMUserQueries folder and modify them when necessary. You can save in your Personal Queries folder any public query that you modify.

Do not run queries in the UCMSystemQueries folder; they are intended to be run by the ClearCase system only.

This section describes the following queries:

- MyToDoList
- UCMCustomQuery1
- Other queries

In addition to the queries included in the database schema, you and your project manager can create your own queries.

MyToDoList

The MyToDoList query finds UCM-enabled records that match all of the following criteria:

- They are assigned to you.
- They are in a state whose state type is Ready or Active.

You or your project manager can modify this query.

UCMCustomQuery1

The UCMCustomQuery1 query finds UCM-enabled records that match any of the following criteria:

- In a state whose state type is Ready and match all of the following criteria:
  - Are assigned to you
  - Have not yet been assigned to a UCM project
  - Have not yet been worked on

- In a state whose state type is Ready or Active and match all of the following criteria:
  - Are assigned to you
  - Have been assigned to the UCM project associated with the current view
  - Have not yet been worked on
- In a state whose state type is Active and match all of the following criteria:
  - Are assigned to you
  - Have been assigned to the UCM project associated with the current view
  - Have been worked on already in the stream to which the current view is attached

My Activities in ClearCase Explorer and the Check Out, Check In, and Add to Source Control dialog boxes in ClearCase present the activities that UCMCustomQuery1 finds. Although you can also see this query in the context of a ClearQuest user database, it is not designed to be run outside the context of a ClearCase view.

Your project manager can modify this query, but you cannot.

Other Queries

A schema enabled for UCM also supplies the queries described in Table 1. You can run any of these queries.

Table 1 Other Queries in the Unified Change Management Schema

Query	Description
ActiveForProject	For one or more specified projects, selects all activities in an active state type.
ActiveForStream	For one or more specified streams, selects all activities in an active state type.
ActiveForUser	For one or more specified developers, selects all assigned activities in an active state type.
AllActivitiesinStream	For specific stream, lists all activities regardless of state type.
MyCompletedWork	Lists all activities in a completed state type.
UCMProjects	Selects all UCM-project records in ClearQuest user database.

State Types and State Transitions

Change requests in a ClearQuest user database move through a pattern, or lifecycle, from submission through resolution. Each stage in this lifecycle is called a *state*, and each movement from one state to another is called a *state transition*.

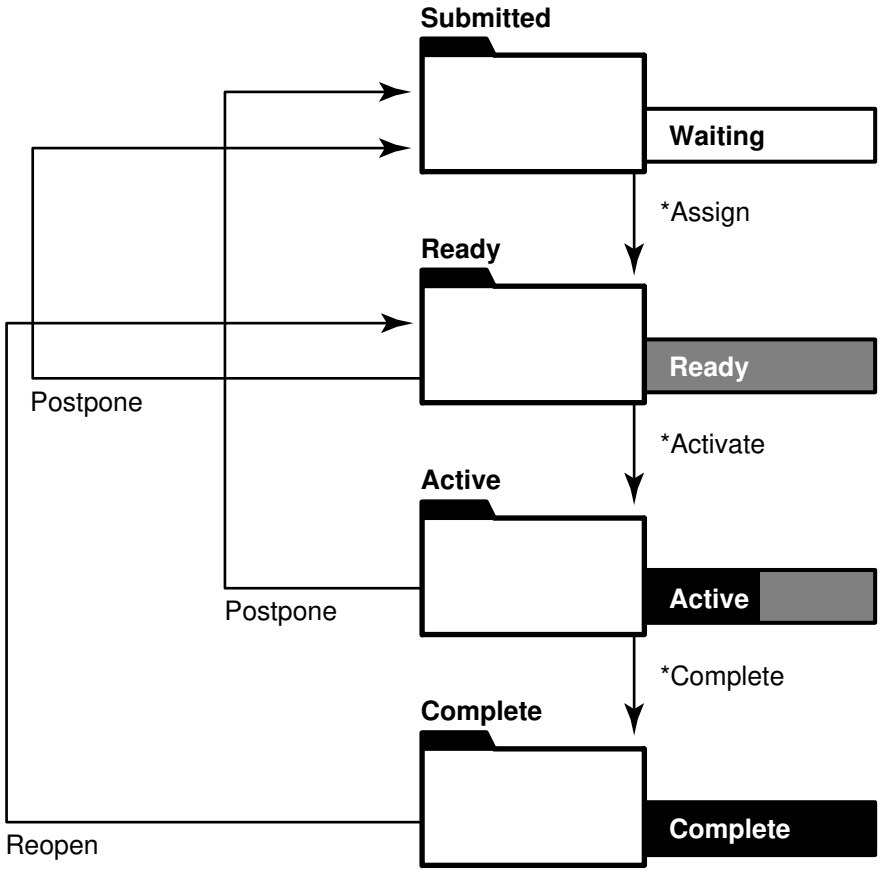
As with record types and records, a state type is a template that defines actions and other attributes associated with a state. The states in a tailored schema must be based on one of the following state types:

- Waiting
- Ready
- Active
- Complete

State Transitions

Schemas include rules for changing records from one state type to the next. Some examples of state transitions are shown in Figure 13.

Figure 13 Example of ClearQuest User Database State Transitions



Some state transitions that records in a schema must follow are:

- **Waiting**. Creating an activity places it in a state based on this state type to indicate, for example, that the work is waiting for someone to resolve dependencies and that it is not ready to be scheduled.
- **Ready**. Assigning an activity usually places it in a state based on this state type to indicate that the work is pending.
- **Active**. Setting an activity in a state based on this state type indicates that the work is in progress.
- **Complete**. When you have completed an activity, you change it to a state based on the Complete state type. Activities in this state do not appear in MyToDoList.

State Types and State Transitions in a Schema Enabled for UCM

Your UCM project manager may give the states in your schema enabled for UCM different names and may create multiple, different states based on the same state type. For example, your schema enabled for UCM may contain the states Scheduled and Deferred, both of which are based on the Active state type but have different associated actions and meanings.

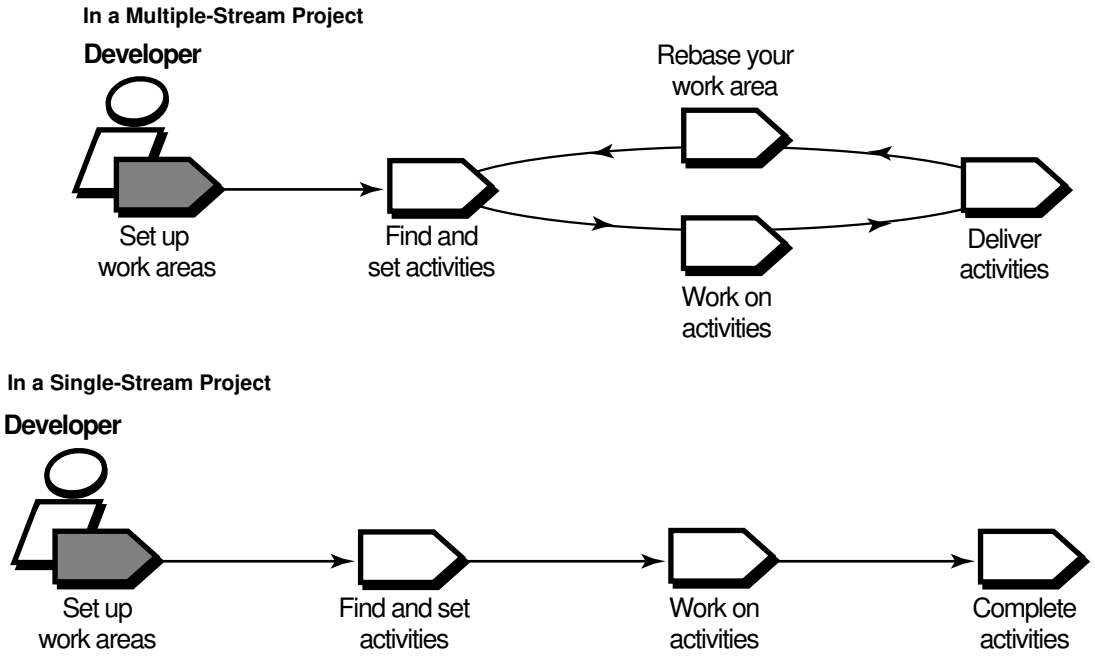
In a schema enabled for UCM, records must follow these state-transition rules:

- **Waiting**. Creating an activity places it in a state based on this state type to indicate, for example, that the work is waiting for someone to resolve dependencies and that it is not ready to be scheduled. Activities in this state do not appear in MyActivities or MyToDoList; they may or may not contain a value in the owner field and may or may not contain a value in the UCM project field. The schema in Figure 13 bases its Submitted and Postpone states on the Waiting state type.
- **Ready**. Assigning an activity usually places it in a state based on this state type to indicate that the work is pending. Activities in this state appear in the activity owner's MyActivities and MyToDoList, but are not part of the owner's stream. The schema in Figure 13 bases its Ready and Re-open states on the Ready state type.
- **Active**. Setting your view to an activity usually places the activity in a state based on this state type to indicate that the work is in progress. Setting your view to an activity also links the UCM-enabled record to the UCM activity. After the UCM activity is linked to the record, you can change the activity record owner but you cannot change the stream that contains the activity. For more information, see *Reassigning Activities* on page 92.

  The **ucm_stream** field in a UCM-enabled record displays the stream to which an activity is linked, and the **ucm_view** field indicates whether an activity is currently set in a view. (Figure 20 on page 64) The schema in Figure 13 bases its Active state on the Active state type.
- **Complete**. When you have completed and delivered an activity, you change it to a state based on the Complete state type. Activities in this state do not appear in MyToDoList, but do appear in **MyActivities** and remain a permanent part of the stream. The schema in Figure 13 bases its Complete state on the Complete state type.

Setting Up Work Areas

2



To set up work areas, you must complete the following tasks:

- Starting the Join Project Wizard
- Choosing a project
- Verifying stream names
- Setting up views
- Choosing components to load into snapshot views
- Accessing your development view
- Connecting to the UCM-ClearQuest integration (only for projects that use the UCM-ClearQuest integration)

For information about the differences in the tasks for single-stream projects, see *Setting Up Work Areas in a Single-Stream Project* on page 58.

Starting the Join Project Wizard

The Join Project Wizard assists you in each step of joining a project. Start Rational ClearCase Explorer and the Join Project Wizard and use this chapter to complete the steps.

To Start ClearCase Explorer

Start ClearCase Explorer by doing one of the following:

- Double-click the Rational ClearCase shortcut on your desktop.
- If you did not install the shortcut, click **Start > Programs > Rational ClearCase > ClearCase Explorer**.

For information about ClearCase Explorer, see *About Rational ClearCase Explorer* on page 25 or click **Help > Using ClearCase Explorer**.

To Start the Join Project Wizard

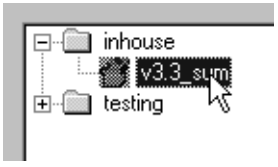
In the ClearCase Explorer Shortcut pane, click **Toolbox**. Then click **UCM > Join Project**.

The Choose a Project dialog box opens.

Choosing a Project

In this page of the Join Project Wizard, choose a project. Navigate the list of PVOBs, folders, and projects and click the project you want to join (Figure 14).

Figure 14 List of Projects



To see a description of a project, select it and click **Properties**.

After you select a project to join, click **Next**. If you join a multiple-stream project, you next verify stream names. If you join a single-stream project, you next set up your integration view (see *Setting Up Your Views* on page 46).

Verifying Stream Names

After you choose a multiple-stream project, this step of the Join Project Wizard opens the Create a Development Stream dialog box, which presents the two streams that Rational ClearCase uses to keep track of your work on this project. Take note of the stream names when the wizard presents them; they are permanent objects that keep track of your work on the project.

Development Stream

If you do not have any streams defined in the selected project, the Join Project Wizard gives your development stream a name based on the following convention: *userID_project-name*. For example, Pat's development stream for the 1.4 release of the Cropcircle project is titled **pat_1.4_cropcircle**. If you want to add a comment to describe the stream that you are creating, click **Advanced Options** and, in the **Comment** box, enter your description. If you have development streams in the selected project, a dialog box lists the stream names. If you want to work on activities in an existing development stream, click **Advanced Options**, click **Use existing development stream**, and select a development stream.

If ClearCase cannot determine your user ID, it substitutes the name of the ClearCase host from which you are creating the stream.

Although you can change the name of your development stream on this step of the Join Project Wizard, we recommend that you do not, unless your project manager has requested it. Following consistent conventions facilitates project management.

Integration Stream

Your project manager creates the integration stream as part of creating the project. This is the stream to which you typically deliver your work and from which you update your work area. The Join Project Wizard presents the name of the project integration stream as a default. The integration stream becomes your parent stream for this project. It is the stream to which your integration view will be attached. This stream also is the default target to which you deliver your work and from which you update your work area.

After you verify your stream names, you set up your views (see *Setting Up Your Views* on page 46).

Choosing a Different Parent Stream

If your project uses feature-specific development streams, you can join the project at the development stream level rather than the integration stream level (see

Feature-Specific Development Streams on page 21). To use one of these parent development streams, click the dropdown list to the right of the name of the default integration stream and select one of the development streams listed.

This development stream, rather than the integration stream, becomes your parent stream for this project. Your integration view is attached to this parent stream. This stream also is the default target to which you deliver your work and from which you update your work area.

After you choose a different parent stream, you set up your views.

Setting Up Your Views

In the next steps of the Join Project Wizard, you set up your development view and integration view (multiple-stream project) or simply set up your integration view (single-stream project) by specifying the following information:

- Which type of view to create for your development view and integration view
- Locations for snapshot views
- Drive mapping for a dynamic view folder (optional)
- Locations for dynamic view storage directories

We recommend that you accept the defaults the Join Project Wizard presents until you are familiar with UCM. For example, the default naming conventions for your views are intended to distinguish your development view from your integration view.

Determining the View Types

As described in *Streams and Views* on page 16, you can use either a snapshot view or a dynamic view to create a directory tree of source files. Your project manager determines the default view types that the Join Project Wizard presents.

Work in a snapshot view when any of these conditions is true:

- Your computer does not support dynamic views.
- You want to optimize build performance to achieve native build speed.
- You want to work with source files under ClearCase control when you are disconnected from the network that hosts the VOBs.
- You want to access a view from a computer that is not a ClearCase host.
- Your development project does not use the *build auditing* and *build avoidance* features of ClearCase.

Work in a dynamic view when any of these conditions is true:

- Your development project uses build auditing and build avoidance.
- You want to access elements in VOBs without copying them to your computer.

Because a dynamic view can show changes in its attached stream at all times without requiring an update, we recommend that you use dynamic views for your integration views whenever they are available.

The two view types behave slightly differently, so remember the type of view you create.

Your project manager, or someone else in your organization, determines the types of views the Join Project Wizard uses for a default when you first join a project. The default is typically based on network configuration and performance needs. For a detailed comparison of snapshot views and dynamic views, see the Help and the *Administrator's Guide* for Rational ClearCase.

Locations for Snapshot Views

When creating a snapshot view, you must specify a view directory into which ClearCase loads, or copies, versions of source files from VOBs into the snapshot view at the directory you specify.

When choosing a directory, consider these constraints:

- The view root directory must be located on a disk with enough space for the files loaded into the view and any other files you add.
- Your organization may restrict where you can create a view. For example, you may be required to use a disk that is part of a data-backup scheme.
- If you want to access the view from other computers, it must be located in a shared directory.

If your makefiles or other files require absolute paths with a drive, you can map the folder for a view to a specific drive. See *Assigning a Snapshot View to a Drive* on page 139.

Under the Hood: A Snapshot View Storage Directory

Every snapshot view has a *view storage directory* in addition to the directory tree of source files that it loads from VOBs. ClearCase uses the view storage directory to keep track of such information as which files are loaded into your view and which versions are checked out to it. The view storage directory is for ClearCase administrative purposes only. Do not modify anything in it.

For every 1,000 elements loaded into the view, ClearCase uses about 400 KB of disk space for the view storage directory.

Locations for Snapshot View Storage Directories

Usually, your ClearCase administrator sets up a storage location, which is a directory on a ClearCase server host on UNIX or Windows, and by default ClearCase locates snapshot view storage directories in the storage location. If your ClearCase administrator sets up more than one storage location, ClearCase selects one of these locations as the default location when you create a view.

If your ClearCase administrator does not set up storage locations, ClearCase sets a directory under the root directory of the snapshot view as the default location for the view storage directory. This is called a co-located view storage directory.

You can override the default location. If your administrator sets up multiple storage locations, you can select one explicitly. If your ClearCase host is set up to store view storage directories (which happens when you install ClearCase), you can place the view storage directory under the root directory of the snapshot view. Or you can choose another location.

If you co-locate the view storage directory under the root directory of the view, be aware of the following recommendations:

- Do not choose this configuration if you use the view when disconnected from the network. You can corrupt the data in the view storage directory if you disconnect it from the network while the view *view_server* process is running.
- Make sure that the view storage directory is accessible to any data backup schemes your organization institutes.

If you override the default location and place the view storage directory under a directory other than the root directory of the view, the directory must follow these conditions:

- Be below a shared network directory on a ClearCase host (for example, a computer running Windows NT or Windows 2000). (A view server process runs on the computer that physically stores the view storage directory or on a host that can access a location on a supported Network Attached Storage (NAS) device which contains the view storage directory, and only a ClearCase host can run a view process.)
- Must remain connected to the network.
- The path for the directory must not use a Windows special share name, for example, the share that is designated by *drive-letter*$ and allows an administrator access to a drive over the network.
- The directory cannot be on a removable storage device or on a laptop.

Note: If you plan to work while disconnected from the network, or if your ClearCase host is not set up to store view storage directories, your administrator must set up storage locations.

To Override the Default Snapshot View Storage Location

1. When creating a view, on the step of the wizard that asks you to choose a location for a snapshot view, click **Advanced Options**.
2. In the Advanced View Options dialog box, do **one** of the following:
   - If **Use Server Storage Location** is selected and your administrator created multiple locations, ClearCase selects one of the locations for you. To choose a different one, click the name of the location.
   - If your computer is set up to hold view storage locations and you want to place the view storage directory in the root directory of the snapshot view or choose another location, select **Use explicit path**.

     Do not select this option if you plan to use the view while disconnected from the network.
3. If you select **Use explicit path**, do **one** of the following:
   - Accept the default path displayed in **View storage location**.
   - Edit the path in **View storage location** to specify a valid location.
   - Click **Browse** and choose a valid location.
4. Click **OK** to return to the view location step of the wizard.

Drive Mapping for Dynamic Views

If your makefiles or other files require absolute paths, map the folder for your view to a drive. When you use a wizard to create a view, ClearCase prompts you to assign the dynamic view to a drive. After creating a view, if you want to change a drive assignment or assign a drive to a team member's view, use the view shortcut menu in ClearCase Explorer to create or modify the assignment.

In addition, you can create or modify drive assignments from Windows Explorer. Any changes you make to a view drive assignments outside ClearCase Explorer will invalidate the view shortcut in ClearCase Explorer.

Path Differences

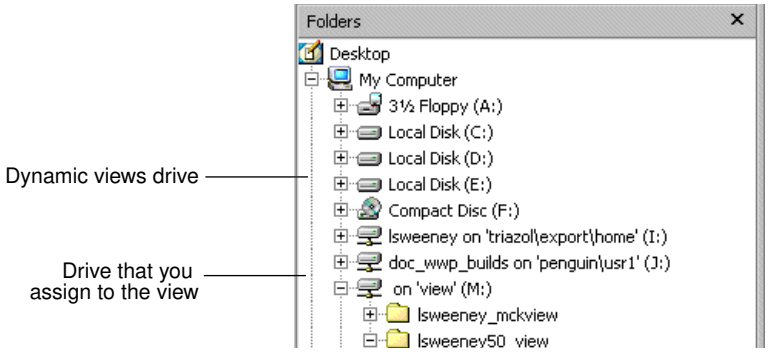
You can access any dynamic view that is started on your computer from the *dynamic-views-drive* (by default, drive M:) in Windows Explorer. However, when you

access a view from the dynamic-views drive, its path includes one more component than when you access a view from an assigned drive (Figure 15).

For example, both of the following paths are available to an element in the **pat_v1.4_cropcircle** dynamic view that is assigned to drive P:

```
M:\pat_v1.4_cropcircle\guivob\prog.c
P:\guivob\prog.c
```

Figure 15 Dynamic Views Started on a Host



Under the Hood: View Storage for Dynamic Views

The first time you create a dynamic view, the wizard provides a default path for the storage directory. Click **Advanced Options** to provide a different path for the view storage directory. The location you choose becomes the default for other dynamic views that you create. Every view has a view storage directory. For dynamic views, ClearCase uses this directory to keep track of which versions are checked out to your view and to store view-private objects. The view storage directory is for ClearCase administrative purposes only. Do not modify anything in it.

The size of the view storage directory depends on the following factors:

- Whether you use the **clearmake** or **omake** *build auditing* and *build avoidance* features
- The size and number of view-private files

For more information, see the **clearmake** and **omake** reference pages in the *Command Reference*.

Valid Locations for Dynamic View Storage Directories

If your computer is set up to store view storage directories, ClearCase reduces communication over the network by locating the view storage directory on your computer.

Consider the following restrictions when choosing a dynamic view storage directory:

- The directory must be a subdirectory of a shared network resource on a ClearCase host (for example, on a Windows NT or Windows 2000 computer) or on a Network Attached Storage (NAS) device. View processes (specifically, *view_server* processes) run on the computer that physically stores the view storage directory or on the computer that can access a location on a supported NAS device which contains the view storage directory. And only ClearCase hosts on Windows NT and Windows 2000 can run view processes.

  The path for the directory must not use a Windows special share name. Special share names usually include the dollar sign ($), such as the *driveletter*$*share-name* that allows an administrator to gain access to a drive over the network. For example, \\bread\c$\view\pat_1.4_cropcircle.vws is not a valid path.

- To maintain data integrity, the view storage directory must remain connected to the network. For example, do not locate the view storage directory on a removable storage device.

- If you locate the view storage directory on a laptop and then disconnect the laptop from the network, all of the following restrictions apply:

  - You cannot use the dynamic view.

  - Team members who try to start your view from their hosts will receive error messages from ClearCase.

  - Any **clearmake** or **omake** process that attempts to wink in a derived object from your view will spend some amount of time trying to contact your view. If it cannot contact your view, it will not consider derived objects in your view as *winkin* candidates for 60 minutes. (You can change the amount of time by setting the CCASE_DNVW_RETRY environmental variable.) For more information, see the **clearmake** reference page.

- If your ClearCase administrator sets up storage locations (which are directories on ClearCase server hosts), you can locate your dynamic view storage directory in a storage location. However, for best performance, we recommend that you locate dynamic view storage directories on your local host.

We recommend that you make the view storage directory accessible to any data backup schemes that your organization institutes.

To Choose a Location for a Dynamic View Storage Directory

1 On the step of the wizard that asks you to specify a view tag, click **Advanced Options**.

2 In the Advanced Options dialog box, do **one** of the following:

- Select **Use Explicit Path** and provide a UNC path to a shared directory on a ClearCase host on Windows NT or Windows 2000. For best performance, we recommend this option.

  The first time you create a view with the wizard, ClearCase presents a template: `\\`*current-hostname*`\<Share>\`*view-tag*`.vws`

  To locate the view storage directory on your computer (recommended), replace `<Share>` with the name of a shared directory on your computer. For example, \\bread\view_storage\pat_v1.4_cropcircle.vws

- Select **Use Server Storage Location**. If your administrator created more than one location, ClearCase selects one for you. You can choose a different one if you prefer.

Under the Hood: Activating VOBs for a Dynamic View

Dynamic views provide transparent access only to the VOBs that are active on your computer. ClearCase activates the VOBs associated with your project when you finish the Join Project Wizard, each time you log on, and when you start ClearCase Explorer. You can deactivate VOBs that you do not use frequently and reactivate VOBs. For information about activating and deactivating VOBs, see *Activating or Deactivating VOBs for Dynamic Views* on page 138.

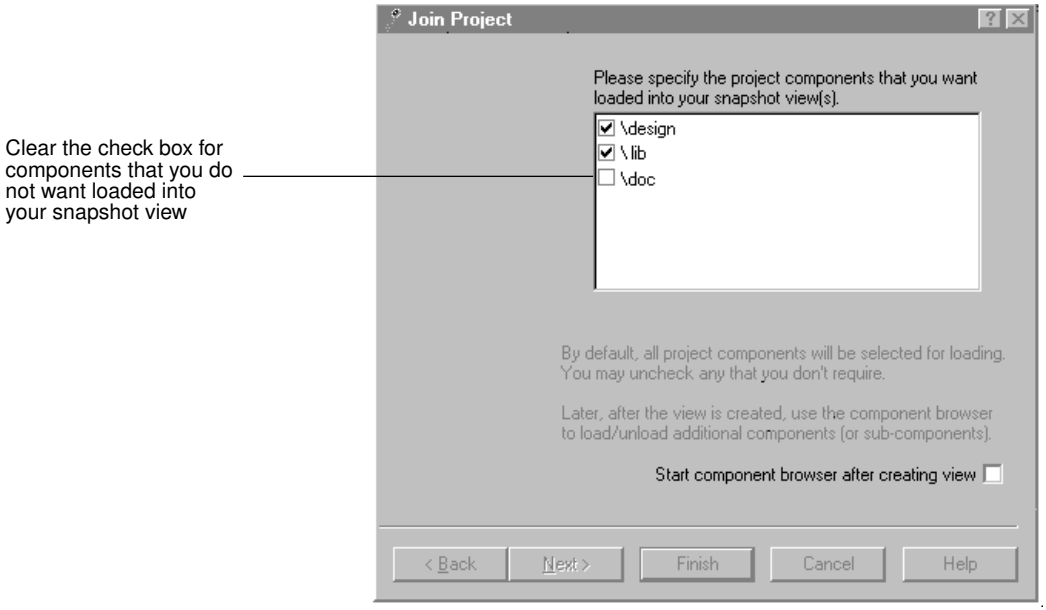
Choosing Components to Load into Snapshot Views

If you choose to create any snapshot views, the Join Project Wizard prompts you to choose the components to load.

Refining the List of Source Files

To save disk space, to reduce the time needed for the initial loading operation, and to reduce time needed for rebase operations, clear the check box for any components that you do not need to complete your work (Figure 16).

Figure 16 Clear Check Boxes for Unused Components



To further refine the list of files and directories that are loaded into your view, select **Start component browser after creating the view**. With the Component Browser, you can specify individual directory and file elements to load.

UNIX VOBs and Case-Sensitivity

If you are selecting elements from VOBs located on a UNIX host, you may encounter problems that are due to case-sensitivity. Because native file systems for UNIX are case-sensitive, it is possible to create two elements from a UNIX host whose names differ only in capitalization. For example, in a UNIX VOB, these two elements are distinct:

```
/design/func_specs/bas
/design/func_specs/Bas
```

However, Windows does not support case-sensitive file lookups and does not distinguish the two elements in the previous example. If you were to load these two elements, only one of them would have the correct data when copied into the view; duplicated files are reported as *hijacked*. (Hijacked files are discussed throughout Appendix A, *Working in a Snapshot View Without the Network*.)

The Rational ClearCase help discusses case-sensitivity issues. We suggest that you NOT use mixed-case names for elements that you store in VOBs.

Loading Elements

After you choose components to load into the view, the Join Project Wizard displays your choices. When you confirm your choices and finish the wizard, ClearCase loads the views with the elements you selected. As ClearCase loads files and directories into each view, it shows a progress indicator in the Snapshot View Update dialog box. The complete list of files copied into your view appears in the Snapshot View Update window.

Accessing Your Development View

ClearCase Explorer provides access to the projects that you join and the views that you create with the Join Project Wizard or View Creation Wizard (see *ClearCase Explorer Navigation* on page 28).

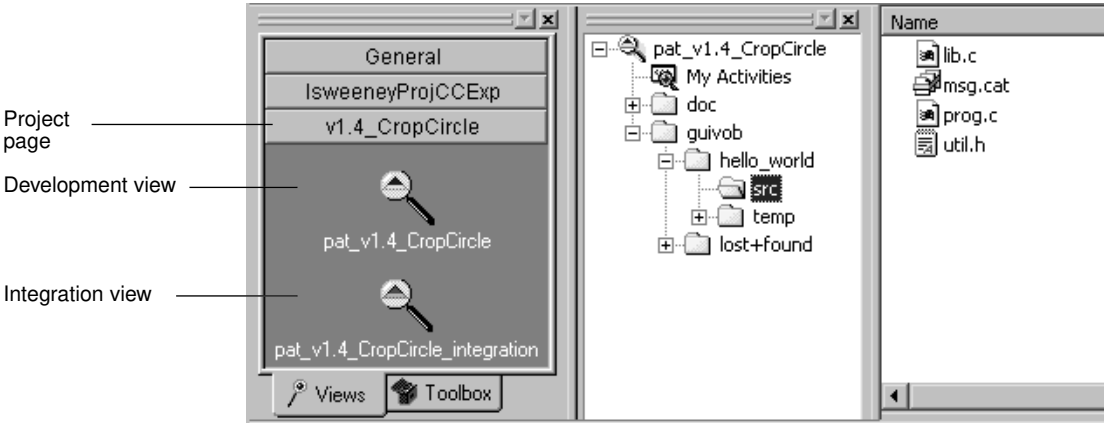
To access your development view, see the following sections:

- *Accessing Your Development View from ClearCase Explorer* on page 54
- *Accessing a Snapshot View from Windows Explorer* on page 55
- *Accessing Dynamic Views from Windows Explorer* on page 56

Accessing Your Development View from ClearCase Explorer

The **Views** tab in ClearCase Explorer Shortcut pane provides access to your projects views (Figure 17).

Figure 17 Development View in ClearCase Explorer



When ClearCase Explorer first starts, it creates on the **Views** tab a page for each project that you join and places view shortcuts on the associated project page. If ClearCase Explorer is running when you join a project, click **View > Refresh View Shortcuts**. A

page for the new project is added on the **Views** tab, and the view shortcuts are added under the project page. Click the project page to open the Shortcut pane and click the shortcut to start the view. By convention, your development view is named *username_project-name*, and your integration view is named *username_project-name*_**integration**.

In a UCM project, ClearCase activates all VOBs that contain components for any of the projects you have joined. For more information, see *Activating or Deactivating VOBs for Dynamic Views* on page 138. If you want to access files in other VOBs, you must activate those VOBs (see *Under the Hood: Activating VOBs for a Dynamic View* on page 52). For more information, see Help.

Starting the view also starts a *view_server* process, which maps the data in the VOBs that are activated on your computer into a directory tree. VOBs that you activate appear as subdirectories of a view. Then you browse a VOB as you would any other directory.

The ClearCase Explorer title bar shows the location of the view in your computer namespace. Any command that you issue from ClearCase Explorer that requires a path uses the path displayed in the title bar. To assign a different path to a view (for example, to change the drive that is mapped to a folder for a view), modify the view shortcut. You do not have to map the folder for a view to a drive.

To access a team member's view from ClearCase Explorer, add a view shortcut to the **Views** tab. For information about creating or modifying shortcuts in ClearCase Explorer, see Help in the ClearCase Explorer.

To access a dynamic view that was created from a UNIX host, first use the Region Synchronizer to import the view tag into your Windows network region. Then, in ClearCase Explorer, click **View > Refresh View Shortcuts**; the shortcut is added under the **General** page in the **Views** tab. Click the shortcut to start the view.

If you want to perform ClearCase operations in a view that you do not own without adding a shortcut to the view in ClearCase Explorer, see *Registering a Snapshot View* on page 142.

Accessing a Snapshot View from Windows Explorer

A snapshot view is a directory tree in a standard file system (plus some hidden, administrative files). You can access it through Windows Explorer as you would access any other directory tree in a file system. To assign a snapshot view to a drive, see *Assigning a Snapshot View to a Drive* on page 139.

Accessing Someone Else's Snapshot View from Windows Explorer

You can access someone else's snapshot view as you would access any other directory on another computer. If the directory owner has shared the directory and set up the proper permissions, you can use Network Neighborhood to access the view.

To use someone else's snapshot view in a ClearCase operation (for example, in a deliver or rebase operation), register the view (see *Registering a Snapshot View* on page 142).

Accessing Dynamic Views from Windows Explorer

You can access any view that you have started on your computer from the *dynamic-views-drive* (by default, drive M:) in Windows Explorer. If you assign the folder for the view to a drive, you can also access the view from the drive in Windows Explorer. For information about assigning dynamic views to drives, see *Drive Mapping for Dynamic Views* on page 49.

Starting Dynamic Views and Activating VOBs

To access a dynamic view from Windows Explorer, you must start the view. If you assign the folder for a view to a drive, ClearCase starts the view when you log on to your computer.

Starting the view also starts a *view_server* process, which maps the data in the VOBs that are activated on your computer into a directory tree. VOBs that you activate appear as subdirectories of a view. Then you browse a VOB as you would any other directory.

To Start a Dynamic View from Windows Explorer

To start the view, navigate to the view in your network folder and expand the view. For example, on Windows 2000:

1. Click **My Network Places > Entire Network > ClearCase Dynamic Views > All Views**.
2. Navigate to the folder for the view and expand the folder. When you select a folder in the view, the UNC path appears in the title bar.

If you want to access files in other VOBs, you must activate the VOBs. For more information, see *Activating or Deactivating VOBs for Dynamic Views* on page 138 and ClearCase Help. If you start a view for a UCM project, ClearCase activates all VOBs for components for that project.

If you are unable to start a dynamic view that is on another host, check with your administrator to make sure that you can access the view storage directory for the view. For more information, see the *Administrator's Guide* for Rational ClearCase.

To stop the view, right-click the folder for the view, and click **ClearCase > Properties of View**, and, in the Properties dialog box, click **Stop View**.

Connecting to the UCM-ClearQuest Integration

If your project uses the UCM-ClearQuest integration, complete the following tasks to set up your computer for the integration:

- Connecting to a schema repository enabled for UCM. (Your ClearQuest user database administrator may have done this for you.)
- Logging on to a ClearQuest user database.

To Determine Whether Your Project is Enabled for ClearQuest

Do the following:

1. In the ClearCase Explorer Shortcut pane, click **Toolbox**. Then click **UCM > Project Explorer**.
2. In Project Explorer, select your project and click **File > Properties**.
3. In the project Properties dialog box, click **ClearQuest**. If **Project is enabled for ClearQuest** is selected, the project uses ClearQuest user databases with ClearCase operations.

Connecting to a Schema Repository Enabled for UCM

A schema repository is a master database that contains data associated with ClearQuest schemas. To use the UCM-ClearQuest integration, you must connect ClearCase to a schema repository that contains UCM-specific data. (For more information, see *The UCM-ClearQuest Schema* on page 34.)

Your ClearQuest user database administrator can create a database profile, which simplifies the process of connecting. If such a profile exists, you must know its name and location. If there is no profile, you must know the properties of the database designated as the schema repository (location, name, and vendor type, as well as password information, and so on).

To Connect to a Schema Repository

1. On your Windows desktop, click **Start > Programs > Rational Software > Rational ClearQuest > Maintenance Tool**.
2. In the Maintenance Tool dialog box, do one of the following:
   - If your ClearQuest user database administrator created a database profile, click **File > Import Profile**. Under **Import Profile Information** in **File Name**, enter the path to the profile or click **Help**. Then click **Next**. Specify your connections and click **Finish**.
   - If you have to specify your connection, click **Connection > New**. Under **Schema Repository Properties**, in **Vendor** select the database vendor for the repository. Enter the required information (each vendor has different information). For more information, click **Help**. Then click **Finish**.
3. In the Maintenance Tool, click **Exit**.

Logging On to a ClearQuest User Database

The first time you perform a ClearCase operation that involves the UCM-ClearQuest integration, such as setting your view to an activity, a prompt appears asking you to enter your ClearQuest user ID and password.

ClearQuest keeps a database of user IDs and passwords. (ClearQuest can use your standard network user ID and password.) Make sure your ClearQuest user database administrator has added a user ID and password to the ClearQuest user database database for you.

When you are prompted, be sure to enter your user ID in this dialog box; ClearQuest stores your login information as part of your Windows user account so it does not need to prompt you again during ClearQuest operations.

Setting Up Work Areas in a Single-Stream Project

The tasks that you perform while setting up work areas in a single-stream project are similar to those for a multiple-stream project. However, there are a few differences in the details.

- You do not verify stream names.

  You cannot create development streams in a single-stream project. Therefore, you create only one view that is attached to the integration stream, the only stream in the project.

- You create only an integration view when you run the Join Project Wizard.

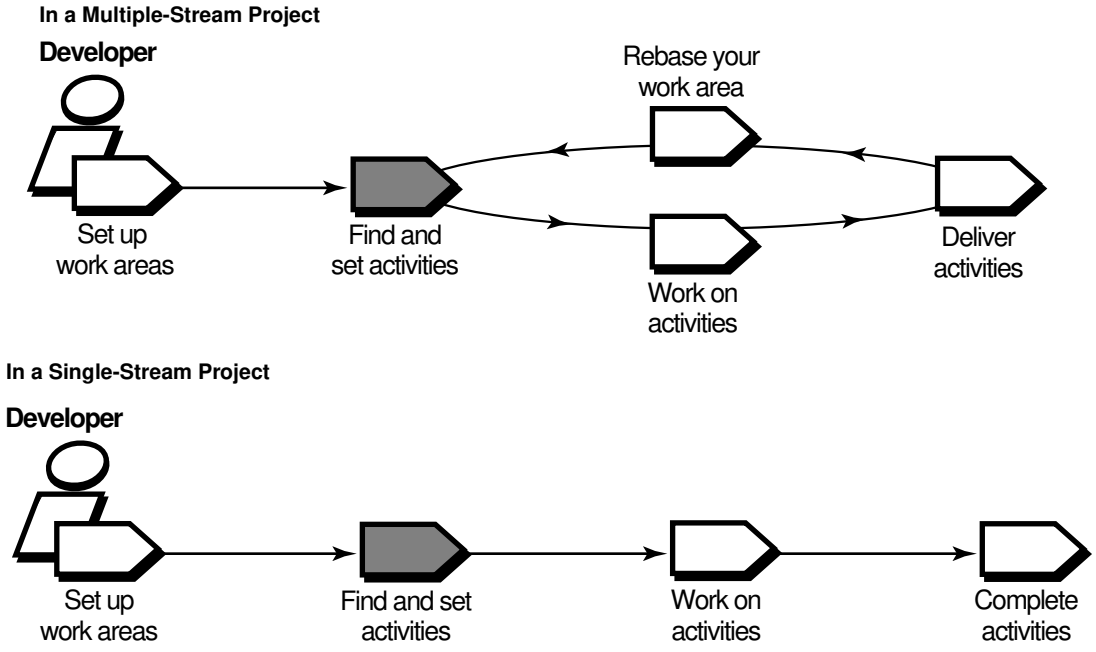
  In single-stream projects, the integration stream is the sole project work area. Each developer has a view (an integration view) in which to work privately. Your integration view is attached to the integration stream, the shared work area for the project.

- You access your integration view rather than your development view.

  In a single-stream project, you have no development views. However, to access your integration view, you follow the same steps as you would to access development views (see *Accessing Your Development View* on page 54).

Finding and Setting Activities

3



Before you start work on the project, find any *activities* that your project manager or other team members assigned to you; if necessary, add new ones to indicate the scope and amount of work scheduled for the project. This chapter describes the following tasks:

- Finding activities
- Creating activities
- Assigning activities in a project that is enabled for ClearQuest
- Setting activities

The descriptions in this chapter apply to both multiple- and single-stream projects. However, for single-stream projects, substitute the terms integration view and integration stream where the text refers to development view and development stream.

Note: Rational ClearCase provides a Web interface through which you can complete some of the tasks described in this and subsequent chapters (see *Introduction* for Rational ClearCase). Ask your system administrator how to access the Web interface. For more information, see the Web interface Help.

Finding Activities

UCM provides several locations from which to find activities:

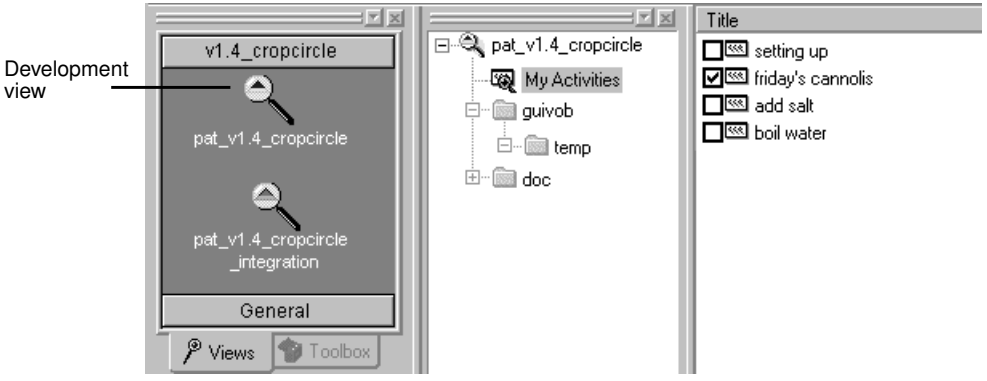
- The **My Activities** folder in ClearCase Explorer
- MyToDoList and other queries (only for projects that are enabled for ClearQuest)
- Project Explorer
- **cleartool lsactivity**
- ClearCase dialog boxes

Finding Activities with the My Activities Folder

In ClearCase Explorer, **My Activities** shows the activities in your work area (Figure 18).

In a project that is enabled for the UCM-ClearQuest integration, **My Activities** shows the results of the UCMCustomQuery1 query. (For more information, see *UCMCustomQuery1* on page 38.) In a project that does not use the UCM-ClearQuest integration, **My Activities** shows the list of UCM activity objects that are in your stream.

Figure 18 My Activities in ClearCase Explorer



To Open My Activities

1 In ClearCase Explorer (see *To Start ClearCase Explorer* on page 44), click the **Views** tab.

2 Click the page for your project. Then select your development view.

3 In the Folder pane, click **My Activities**.

Finding Activities with MyToDoList and Other Queries

If your project is enabled for the UCM-ClearQuest integration, you can use the ClearQuest client to find activities.

To Start the ClearQuest Client

On the Windows desktop, do one of the following:

- Double-click the Rational ClearQuest shortcut on the desktop.
- If you did not install the shortcut, in the task bar, click **Start > Programs > Rational Software > Rational ClearQuest > ClearQuest**.

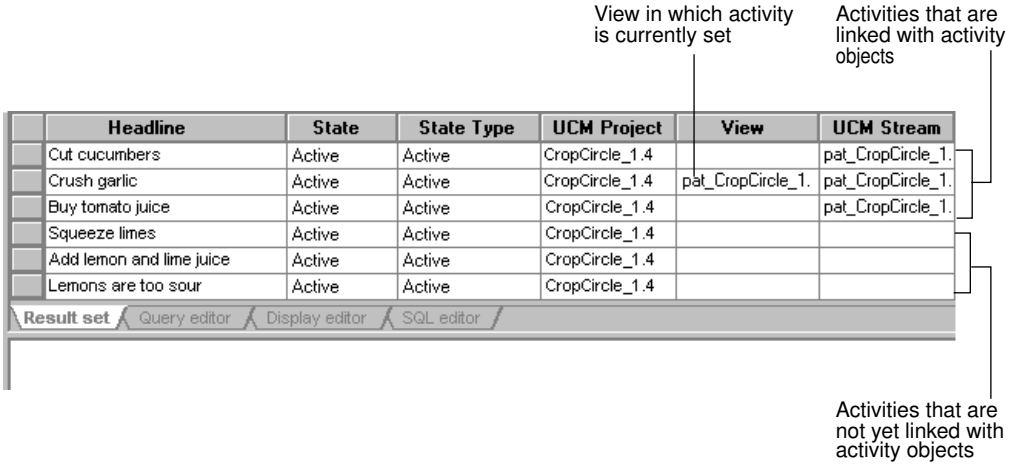
The login dialog box appears. For information about connecting to ClearQuest, see *Connecting to the UCM-ClearQuest Integration* on page 57.

When you are logged in to the ClearQuest user database, the ClearQuest client window appears. For information about the ClearQuest client, see *Display of ClearQuest User Database Information* on page 32 and Help.

Using MyToDoList

The MyToDoList query finds all UCM-enabled records that are assigned to you in a ClearQuest user database, even if they have not been linked with activity objects or if they have been linked with activity objects in someone else's stream (Figure 19). For more information, see *Queries, Reports, and Charts* on page 37 and *UCM-Enabled Record Types and UCM Activities* on page 35.

Figure 19 Linked Activities in MyToDoList Query



To Open the MyToDoList Query

1 In the ClearQuest client, in the left pane, click **Workspace > Public Queries > UCMUserQueries**. Then run the MyToDoList query.

The query output displays in the Query results pane (Figure 19).

2 To arrange the list by information displayed in a column, right-click the column heading and select a sort order from the shortcut menu. For example, to arrange activities alphabetically, right-click **Headline** (the first column heading) and on the shortcut menu, click **Sort Ascending**.

3 To see more information about an item on your to do list, select it. ClearQuest displays the record details in the Record form pane.

Creating Your Own ClearQuest Query

If none of the existing queries suits your needs, you can create a new one. When creating a query to find UCM-enabled records, note the following:

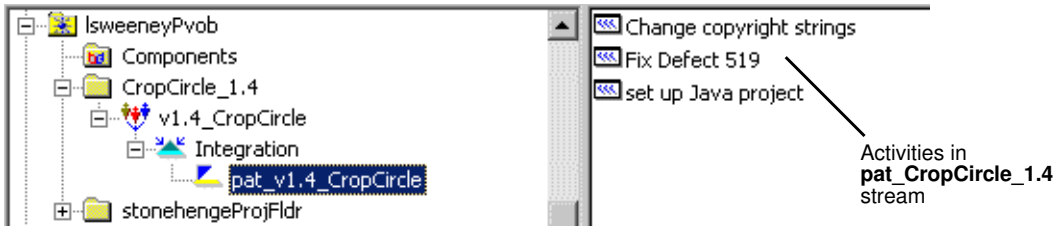
- To search across all UCM-enabled record types, in the Choose a Record dialog box, select **All_Ucm_Activities**. This selection limits your query to fields common to all UCM-enabled record types.
- You cannot use a query to find ClearCase change-set information.

For more information about creating a query, see Help.

Finding Activities with Project Explorer

The Project Explorer arranges activities hierarchically by stream and by project (Figure 20).

Figure 20 Activity List in Project Explorer



To start the Project Explorer, in the ClearCase Explorer Shortcut pane, click **Toolbox**. Then click **UCM > Project Explorer**.

The Project Explorer details pane shows only the UCM activity objects that are in the currently selected stream.

In a project that is enabled for ClearQuest, each activity in a stream is linked to a UCM-enabled record in the associated ClearQuest user database for the project. For more information, see *UCM-Enabled Record Types and UCM Activities* on page 35.

Finding Activities with cleartool lsactivity

You can use **cleartool lsactivity** to find UCM activity objects in streams. This section describes how to find activity objects in your development stream. For information about finding other activity objects with this command, see the **lsactivity** reference page in the *Command Reference*.

To List the Activity Objects in Your Development Stream

1 Access your development view. (For more information, see *Accessing Your Development View* on page 54.)

2 Enter the following command:

cleartool lsactivity –cview

In a project that is enabled for ClearQuest, each activity displayed is linked to a UCM-enabled record in an associated ClearQuest user database. For more information, see *UCM-Enabled Record Types and UCM Activities* on page 35.

Finding Activities with ClearCase Dialog Boxes

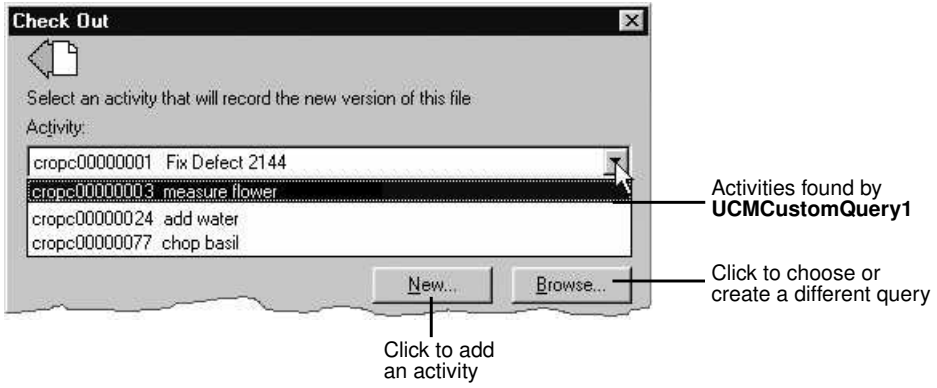
During the checkin, checkout, and add-to-source control operations, a dialog box opens for you to set an activity and optionally enter a version creation comment.

If the view from which you start the dialog box currently has an activity set, the Activity box shows the headline of the activity. If there is no activity set, the Activity box is blank. Use the list to select one, click **New** to create an activity to set (see *Creating UCM Activities* on page 66), or click **Browse** to find an activity to set.

If the project is enabled for ClearQuest, the dialog box displays the IDs of activities found by the UCMCustomQuery1 query (Figure 21). (For more information, see *UCMCustomQuery1* on page 38.)

In projects that do not use the UCM-ClearQuest integration, the list shows the IDs of UCM activity objects in the stream to which is attached the view from which you issued the command. Click **Browse** to show the activities that other team members created in your stream.

Figure 21 Check Out Dialog Box for a Project Enabled for ClearQuest



Creating UCM Activities

Your organization may have policies regarding the scope of a UCM activity. For example, these policies may require that all activities refer to a change request in your change-request management system. The process for creating activities differs, depending on whether your project uses the UCM-ClearQuest integration.

This section describes the following tasks:

- Creating activities in a project that is enabled for ClearQuest
- Creating activities in a project that is not enabled for ClearQuest

Creating Activities in a Project Enabled for ClearQuest

In a project that is enabled forClearQuest, an *activity* usually refers to two objects that are linked together: a UCM activity object and a UCM-enabled record in a ClearQuest user database. The process of creating an activity in a project that is enabled for ClearQuest entails the creation of two objects in a specific order:

1. You create records (based on a UCM-enabled record type) in a ClearQuest user database. You can create records from ClearCase Explorer, the ClearQuest client, or ClearCase dialog boxes.
2. When you set a view to an activity (which you can do by clicking **Actions > WorkOn** for a ClearQuest user database record), ClearCase does the following:
   - Creates an activity object in the stream to which the view is attached and
   - Links the activity object to the UCM-enabled record in ClearQuest.

   You do not create UCM activity objects directly.

To Create Activities from the ClearQuest Client

Note: You can create and use a template of default values for the Submit form in the ClearQuest user database. Such a template can save time and be helpful when you are entering multiple, related records. For information about using a template, see Help.

1. Start the ClearQuest client (see *To Start the ClearQuest Client* on page 63).
2. In the ClearQuest client window, click **Actions > New**.
3. In the Choose a Record Type dialog box, select an appropriate UCM-enabled record type. (Your project manager sets up the record types that are in your ClearQuest user database.) Then click **OK**.

   ClearQuest highlights in red the required fields and the fields that contain invalid values. If a field contains an invalid value, you can right-click the field and then click **Error Message** to see any message or explanation that your project manager may have included. If you have questions about what type of information a field requires, right-click the field and then click **Help**.
4. Complete the required fields in the ClearQuest user database Submit form.
5. In the **UCM Project** list, select your project. (Your project manager determines the location for this list when she creates the corresponding record type.) Then click **OK**.

After you create an activity from the ClearQuest user database, you must complete an additional procedure to make the activity appear in MyToDoList or MyActivities. (See *To Make Activities Appear in My Activities and MyToDoList* on page 69.)

To Create Activities from ClearCase Dialog Boxes

In a project enabled for ClearQuest, this procedure accomplishes these tasks:

- Creates a UCM-enabled record in the ClearQuest user database and assigns the activity to you.
- Creates an activity object in the stream to which the view is attached and links the activity object to the UCM-enabled record in the ClearQuest user database.
- Sets your view to the activity.

To create activities:

1. In the Check Out, Check In, or Add to Source Control dialog box (which opens as part of the corresponding operation), click **New**.
2. From the New menu, select a record type. (All record types on the New menu are UCM-enabled.)
3. In the Submit dialog box, complete the required fields, which are highlighted in red.
4. To create a UCM-enabled record and assign the activity to yourself, click **OK**.
5. To add this activity to your stream and set your view to the activity, select it in the Check Out, Check In, or Add to Source Control dialog box and click **OK**.

Creating Activities in a Project Not Enabled for ClearQuest

In UCM projects that do not use the UCM-ClearQuest integration, you create activity objects directly in a specific stream.When you add an activity, it remains in your stream unless you use the **rmactivity** command to remove it.

To Create Activities from ClearCase Dialog Boxes

In a project not enabled for ClearQuest, follow these steps to create an activity when you process an element:

1. In the Check Out, Check In, or Add to Source Control dialog box (which open as part of the corresponding operation), click **New**.
2. In the New Activity dialog box, enter the headline of the activity. The activity ID is automatically generated.
3. Click **OK**.

When you add an activity from the Check Out, Check In, or Add to Source Control dialog box, ClearCase sets your view to the activity.

To Add Activities from the View Properties Dialog Box

In a project not enabled for ClearQuest, add an activity from the view properties.

1. Open a view properties dialog box. For example, in ClearCase Explorer, right-click the root directory of the view, and click **Properties of View**.
2. On the **General** tab of the Properties dialog box, click **Change**.
3. In the Select Activity dialog box, click **New**.
4. In the New Activity dialog box, enter an activity headline and click **OK**.
5. In the Select Activity dialog box, click **OK**. The activity headline appears in the Properties dialog box under **Current default activity for this view**.
6. In the Properties dialog box, click **OK**.

Assigning Activities in a Project Enabled for ClearQuest

The concept of assigning an activity applies only to UCM-enabled records. In general, *assigning* an activity means making a person responsible for some portion of the development of an activity. An activity assignment may change several times during a development cycle. For example, a project manager may take initial responsibility for an activity. When she sets the schedule for a given project, she assigns the activity to a developer on the project. After you begin work on an activity, the activity assignment usually does not change.

ClearQuest user database records include fields to keep track of the current owner of an activity. Your organization may restrict who can change the value in the owner field.

> **Note**: UCM activity objects exist in a particular stream. They record the activity creator, but the creator information cannot be changed, and has no relationship to the concept of activity assignment that is used in ClearQuest.

To Make Activities Appear in My Activities and MyToDoList

By default, the MyToDoList query and My Activities find records that are both in a Ready or Active state type and assigned to you. To make an activity appear in MyActivities or MyToDoList, you must change its state type and owner.

1. In the ClearQuest user database, run a query to find an activity. Then click the activity in the Query editor pane.
2. In the Record form pane, click **Actions** and choose a command that moves the activity to a state based on the Ready state type.

3 In **Owner**, select your user ID.

4 To see the activity in MyToDoList, in the left pane, click **Workspace > Public Queries > UCMUserQueries**. Then run the MyToDoList query.

To see the activity in My Activities in ClearCase Explorer, select your view from the Shortcut pane. Then click **My Activities** in the Folder pane.

Setting Your View to an Activity

To work on an activity, you must set your development view to that activity. ClearCase associates all work in your view to the currently set activity, adding each version you create in your view to the activity change set.

Note: Only one user should set an activity in the same UCM view at the same time. If multiple users attempt to set an activity in the same UCM view at the same time, one user succeeds and other users encounter a set-activity retry error. We recommend that only one person have primary use (make modifications) in a UCM view and other users do read-only actions in the view.

You can set activities from the following locations:

- ClearCase Explorer
- MyToDoList or the results of any other query (only for projects that are enabled for ClearQuest)
- From a command prompt with **cleartool setactivity**
- Dialog boxes that open during checkout, checkin, and add to source control operations

This section also describes the following tasks:

- Unsetting an activity
- Seeing which activity is currently set in the view

Setting Activities from ClearCase Explorer

1 In ClearCase Explorer, click the **Views** tab.

2 Click the page for your project. Then select your development view.

3 In the Folder pane, click **My Activities**.

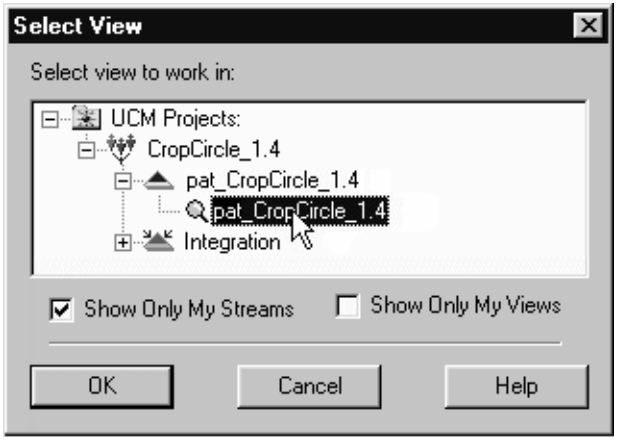
4 In the Details pane, select the check box next to the activity you want to set.

Setting Activities from the ClearQuest User Database

1. Start the ClearQuest client (see *To Start the ClearQuest Client* on page 63) and do one of the following:
   - Display MyToDoList and, in the left pane, click **Workspace > Public Queries > UCMUserQueries** and run the MyToDoList query.
   - Use another query to find the activity to set.
2. In the Query results pane, click the activity to set.
3. In the Record form pane, if it is not already selected, select your project in the **UCM Project** list. (Your project manager determines the location for this list when she creates the corresponding record type.)
4. Click **Actions > WorkOn**.
5. In the Select View dialog box (see Figure 22), select **Show Only My Streams**.
6. Click the plus sign (**+**) next to your stream; then select your development view and click **OK**.

The status bar in the ClearQuest user database indicates that your view is being set to the activity.

Figure 22 Select Your View



To start working on your project source files, navigate to your view and check out the source files.

Setting Activities with cleartool setactivity

In a project that is enabled for the UCM-ClearQuest integration, before you use **cleartool setactivity**, a UCM activity object must exist in the stream, and you must

know the activity record ID. For information about creating an activity object, see *Setting Activities from the ClearQuest User Database* on page 71 and *Creating UCM Activities* on page 66.

To Set an Activity with cleartool setactivity

1. Access your development view. (For more information, see *Accessing Your Development View* on page 54.)
2. Find an activity selector to use. For information about finding an activity selector for a particular activity, see *Finding Activities with cleartool lsactivity* on page 65.
3. Enter the following command:

   cleartool setactivity *activity-selector*

For example, in a project that is not enabled for ClearQuest, enter the following command:

cleartool setactivity changing_copyright_strings

When executed in a view that is associated with a project enabled for ClearQuest, the **setactivity** command takes an activity-selector that is an ClearQuest user database record ID of an existing ClearQuest record. For example:

cleartool setactivity SAMPL123456

You can set only one activity per view at a time. ClearCase associates all checkouts in your view with the currently set activity until you unset the activity (see *Unsetting Your View from an Activity*) or set another one.

For more information, see the **setactivity** reference page in the *Command Reference*.

Setting Activities from ClearCase Dialog Boxes

During the checkin, checkout, and add-to-source control operations, ClearCase opens a dialog box that prompts you to enter a version creation comment and to set an activity.

In a project that is enabled for ClearQuest, the dialog boxes list the headlines of activities found by the UCMCustomQuery1 query. (For more information, see *UCMCustomQuery1* on page 38.) You can click **Browse** to use or create a different query (Figure 21).

In projects that do not use the UCM-ClearQuest integration, the dialog boxes show the headlines of activity objects in the stream to which is attached the view from which you issued the command. You can click **Browse** to see the activities that other team members have created in your stream.

Unsetting Your View from an Activity

You can change any view to have no currently set activities, for example, if you need to remove the view or remove the activity currently set in the view.

To Unset Activities from ClearCase Explorer

1. In ClearCase Explorer, display your work area and click **My Activities**.
2. In the Details pane, clear the check box next to the currently set activity.

To Unset Activities from a Command Prompt

1. At a command prompt, change to any directory in your view.
2. Enter the following command:

   cleartool setactivity –none

To See Which Activity Is Currently Set

Use either ClearCase Explorer or a command window.

From ClearCase Explorer

In ClearCase Explorer, display your work area and click **My Activities**.

The Details pane shows a check mark next to the currently set activity.

From a Command Prompt

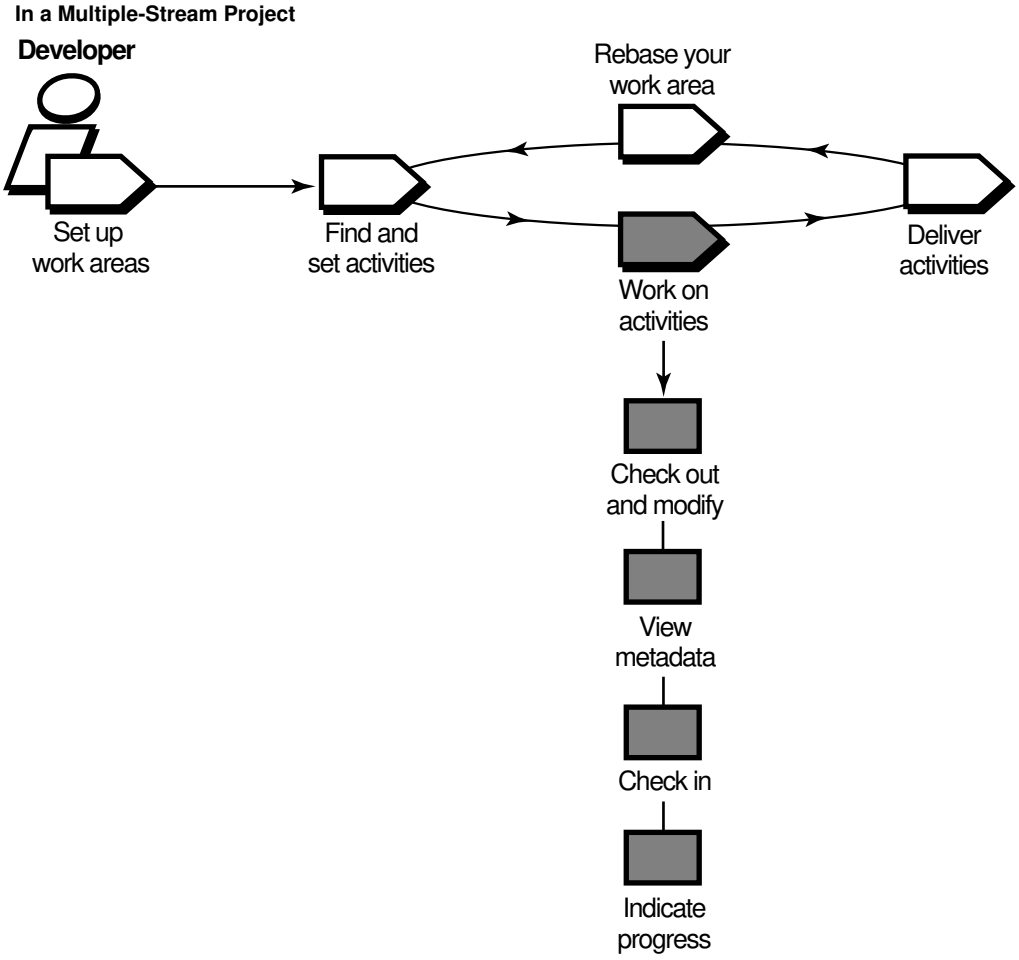
1. At a command prompt, change to any directory in your view.
2. Enter the following command:

   cleartool lsactivity –cview

A message displays the date of creation, the activity headline, and owner of the current activity.

Working on Activities

4



Working on activities involves the following tasks:

- Checking out files
- Working with checkouts
- Canceling checkouts
- Checking in files

- Testing your work
- Indicating your progress

In a multiple-stream project, the work in your development stream is unavailable to other team members until you deliver it (see Chapter 5, *Delivering Activities*).

In a single-stream project, the work in your work area is available to other team members when you check in. For information about working in a single-stream project, see *Working in a Single-Stream Project* on page 95.

Checking Out Files

To modify files under source control, you must check them out and make sure that you have set your view to an *activity*. Checking out adds versions to the activity change set. (Checked-out versions are displayed as part of the activity change set.)

> **Note**: For information about accessing your development view, see *Accessing Your Development View* on page 54.

If your project uses the UCM-ClearQuest integration, your project manager may enable a policy called Perform ClearQuest Action Before Work On. This policy runs a customized ClearQuest user database script when you attempt to work on an activity. For example, your project manager may create a script that checks whether your user name matches the name in the ClearQuest user database record Owner field. Before you can work on the activity, you have to meet all criteria established by the Perform ClearQuest Action Before Work On policy.

To Check Out Files and Set Activities

You can check out files and set activities in a ClearCase context or using a ClearCase shortcut menu.

From ClearCase Explorer

To check out files and set activities in a ClearCase context, use ClearCase Explorer.

1. In ClearCase Explorer, navigate to the directory in your development view that contains the files you want to check out. Then select the files.
2. Right-click one of the selected files and click **Check Out**.

   ClearCase opens the Check Out dialog box with the currently set activity selected in the Activity list.
3. To set a different activity (or, if you have not set the view to an activity), do the following:

a. Select one from the **Activity** list. To filter the list, click **Browse**.

b. If an appropriate activity does not exist, create and set a new one by clicking **New**. If your project is enabled for ClearQuest, select a record-type from the shortcut menu that appears when you click **New**. The activity state changes to an Active state type, and ClearCase adds the version you check out to the activity change set.

4. Describe the modifications you plan to make in the Checkout Comment box. Use comments to describe modifications to individual versions within an activity change set. Your comments are displayed as part of the element history and the version properties.

5. Click **OK**.

For information about checking out VOB links in a snapshot view, see *Symbolic Links in Snapshot Views* on page 80.

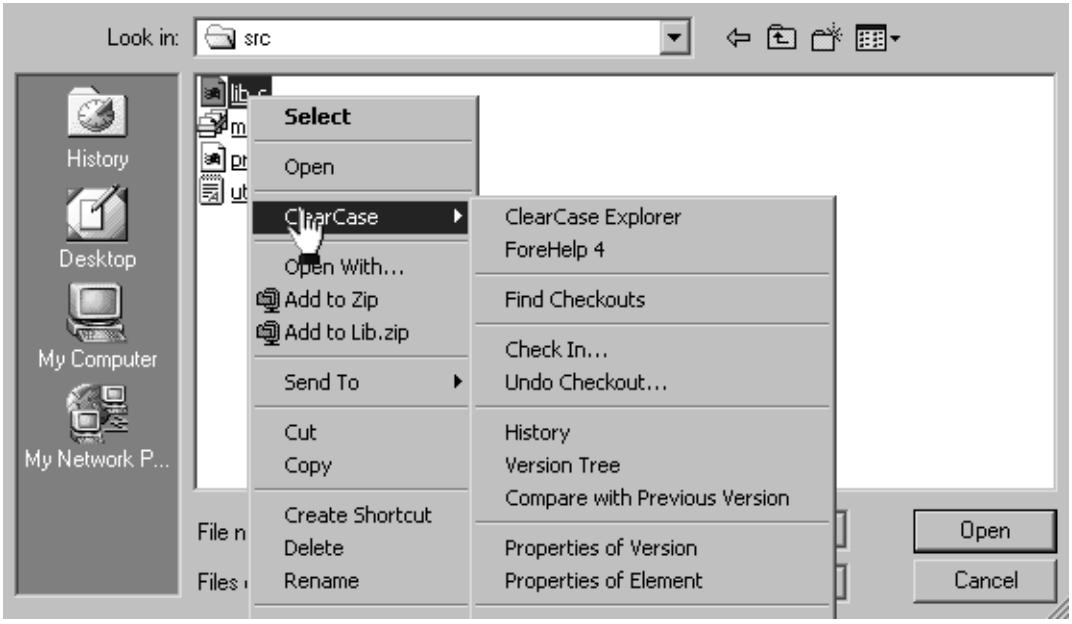
From the ClearCase Shortcut Menu

Access the ClearCase shortcut menu from the Open dialog box that many applications use. To check out a file from an Open dialog box:

1. In an application such as Microsoft Notepad, click **File > Open**.
2. In the Open dialog box, navigate to the file or directory that is under source control.
3. Right-click the file; then click **ClearCase > Check Out**.

As illustrated in Figure 23, right-clicking a loaded version of a ClearCase element in the Open dialog box displays the ClearCase shortcut menu.

Figure 23 The ClearCase Shortcut Menu from the Open Dialog Box



Checking Out Directories

Directories, as well as files, are under ClearCase source control, yet you rarely need to check out a directory explicitly. ClearCase checks out and checks in the parent directory of a file when you add the file to source control.

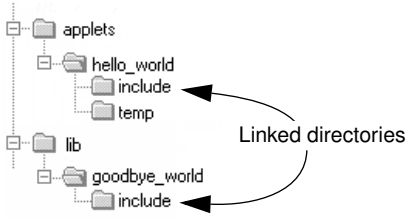
What does it mean for a directory to be under source control? In a version-controlled directory, ClearCase creates a new version of the directory when you add or rename a file element under source control. Having versions of directories can be helpful if, for example, you rename a source file used in a particular release and then modify your makefile to reflect this change. If you need to rebuild a previous release, you can set up your view to select the version of the directory that contains the file under its previous name.

Note: When you issue commands from a command-line interface (CLI), such as an MS-DOS command prompt, ClearCase does not check out directories automatically. When using a CLI to change a directory element, you need to check out the directory explicitly. For more information about checking out files and directories from a CLI, see the **checkout** reference page in the *Command Reference*.

Under the Hood: VOB Links

A VOB link makes a file element or directory element accessible from more than one location in the VOB namespace. There are two kinds of VOB links: *symbolic links*, which are available for file and directory elements, and *hard links*, which are available for file elements only. We recommend that you use VOB symbolic links instead of VOB hard links whenever possible. In Figure 24, the directory element include is linked.

Figure 24 VOB Link



You use the **cleartool ln** command to create VOB links. For more information, see the **ln** reference page in the *Command Reference*.

You cannot successfully link to a component that is not part of your configuration and therefore not accessible to your VOB namespace. The link is created, but you cannot see the target that is outside your configuration. You should ask your project manager to add that component to the project and rebase to that configuration.

Symbolic Links and Hard Links in Dynamic Views

In *dynamic views* (which use the MVFS, or multiversion file system), VOB links behave similarly to symbolic links or hard links in a UNIX file system: symbolic links point to a file or directory element in a different location, and hard links are alternative names for a single file element.

You cannot check out a VOB symbolic link; you must check out the symbolic link target.

When you check out a hard-linked element from a specific path, ClearCase considers other paths for the element as "checked out but removed." That is, to prevent you from modifying the element from multiple paths, ClearCase executes standard checkout behavior at only one path (the one from which you entered the **checkout** command), but does not create view-private files at other paths. For information about standard checkout behavior, see the **checkout** reference page in the *Command Reference*.

In ClearCase Explorer, the shortcut menu for a symbolic link includes the **Symlinks** command, which changes your context to the target (**Warp to Symlink Target**) or provides ClearCase commands to operate on the target without changing your context

(**Symlink Target Operations**). You must mount the VOB that contains the target of the link. Use one of these commands to find and check out the link target. When you check in the modified link target, ClearCase updates all associated symbolic links.

Symbolic Links in Snapshot Views

Snapshot views created from a ClearCase host running Windows software do not support links. ClearCase approximates VOB symbolic link behavior in the following ways:

- If a *load rule* selects a symbolic link, ClearCase copies the link target into the view at the link path.

  We recommend that you do not load a linked directory more than once in your view unless it is necessary for build purposes. Although ClearCase keeps track of multiply loaded elements accurately, you may become confused and modify the wrong copy of a file, losing information upon checkin. For more information, see *Caution: Losing Data Because of VOB Hard Links* on page 81.

- You cannot check out a file element from a symbolic link path; you must check out the link target.

  In ClearCase Explorer, the shortcut menu for a symbolic link includes the **Symlinks** command, which changes your context to the directory containing the target (**Warp to Symlink Target**) or provides commands to perform ClearCase operations on the target without changing your context (**Symlink Target Operations**). Use one of these commands to find and check out the link target. When you check in the modified link target, ClearCase updates all associated symbolic links loaded in your view.

  Unlike directory link targets, a file link target can be checked out only if it is loaded in your view.

- The **Add to Source Control** command (available from ClearCase graphic user interfaces) checks out, modifies, and checks in the link target directory whether you issue the command from a directory symbolic link or from the link target (and whether or not the target directory is loaded into the view).

  If you issue the **cleartool checkout** command from a symbolic link directory, you must use the following syntax:

  cleartool checkout .

  A **checkout** command issued for the current directory checks out the link target whether or not the target directory is loaded into the view.

  However, if you use the **cleartool checkout** *dirname* form of the command to check out a different directory, *dirname* must be a link target.

When either you or ClearCase checks in a link target directory, ClearCase updates the associated symbolic links that are loaded in your view.

- If you *hijack* a file symbolic link, the Update Tool detects it. However, you cannot check out the hijacked symbolic link. To add your hijacked changes to the VOB, you must check out and modify the link target.

Hard Links in Snapshot Views

Each time a load rule selects a hard link, ClearCase loads the element into the view as a standard file.

Caution: Losing Data Because of VOB Hard Links

If you load multiple instances of a hard-linked element into a snapshot view, you must be careful to check out, modify, and check in only one instance of the file. When you check in a hard-linked file (or a file below a symbolic-linked directory), ClearCase updates all other instances in your view, which could result in loss of data if you modified multiple copies of the same file. (Note that, when updating instances of files because of a checkin, ClearCase renames any *hijacked* file to *filename*.keep before updating it.)

For example, the following sequence of events will lead to lost data:

1. You check out the hard-linked file src\util.h.
2. ClearCase removes the read-only attribute from util.h in the src directory only (which is the location from which you issued the **checkout** command).
3. You modify src\util.h but do not check it in.
4. Later, you lose track of which file you checked out. You then remove the read-only attribute and modify util.h in the temp directory.
5. You check in temp\util.h. Even though you checked out and modified src\util.h, ClearCase does not prevent you from checking in temp\util.h; with a VOB hard link, temp\util.h is just another name for src\util.h.
6. Any changes you made to src\util.h are lost upon checkin because ClearCase updates all copies of duplicated files when you check in an element. Note that ClearCase does not consider any copy of util.h to be hijacked (even if you change attributes), because you checked out the element in the VOB.

Resolving Checkout Problems

You can encounter problems when attempting to check out an element. Table 2 suggests ways to resolve common problems.

Table 2 Resolution of Checkout Problems

Problem	Likely cause	Suggested resolution
Snapshot view is out of date	You are checking out a nonexplicit version, and the snapshot view is not up-to-date with the VOB. If the element being checked out were updated, a different version would be loaded.	Update the element and then check out the updated version.
Target branch is already reserved	Another view holds a reserved checkout of the target branch.	Check out the branch as unreserved.
Non-**LATEST** version selected	You are checking out a nonexplicit version, the config spec selects a version that is not the **LATEST** on the branch, and there is no **mkbranch** rule. This can occur because a label or a time rule selects elements.	Check out the **LATEST** version.
File is hijacked and does not correspond to the selected version	The file is in the `hijacked/nocheckout` state or the checkout explicitly specifies a different version.	Check out the hijacked file; or merge the hijacked version and the selected version, and use the merge result as the checkout data.
A **mkbranch** rule in the view config spec fails and automatic branch creation fails	The branch exists. The config spec does not select the branch before specifying that the branch should be created (as it should). Or, for a dynamic view, a time rule prevents it from seeing the branch. Or, for a snapshot view, the branch was created after the view was lasted updated.	Check out the **LATEST** version on the branch; for an out-of-date snapshot view, this means an update of the element followed by a checkout of **LATEST**.

Working with Checkouts

After you check out a file, you do not need to use ClearCase until you are ready to check it in. However, you may want to use some ClearCase tools for the following tasks:

- Viewing element history
- Comparing versions of elements
- Tracking checked-out versions
- Viewing an activity change set
- Moving versions to a different activity

Viewing Element History

The History Browser displays the history of modifications to an element, including version-creation comments (entered when someone checks out or checks in an element).

To View Element History

In ClearCase Explorer, right-click an element and click **History**.

Comparing Versions of Elements

As you modify source files, you may want to compare versions to answer such questions as these:

- What changes have I made in my checked-out version?
- How does my checked-out version differ from a particular historical version or from the version that another developer is using?

To Compare with a Predecessor

In ClearCase Explorer, right-click a file and click **Compare with Previous Version**.

To Compare with a Version Other Than the Predecessor

1. In ClearCase Explorer, right-click a file and click **Version Tree**.
2. In the Version Tree Browser, right-click a version and click **Compare > with Another Version**.

   The cursor changes to a target icon.

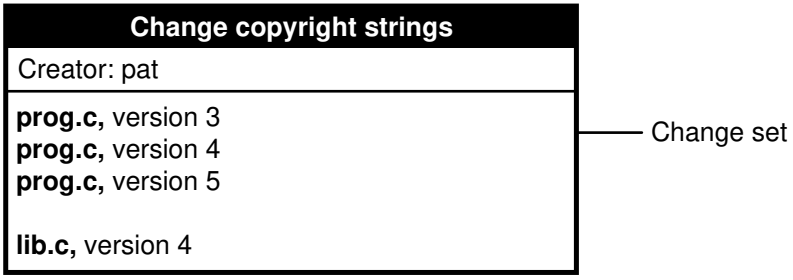
3 Click the version you want to compare.

Comparing with a Change-Set Predecessor

A change set lists all versions you have created while working on a specific activity. In some cases, you create more than one version of an element for a specific activity. For example, for the activity **Change copyright strings** you check in a version of prog.c and realize later that you missed a string in the file. You then set your view to the activity **Change copyright strings**, check out prog.c, modify the string, and check in the file.

To see the changes you have made to an element while working on an activity, you can compare a version in the activity change set with the version that immediately precedes the change set. In Figure 25, the change set predecessor of prog.c is version 2. You can compare any version of prog.c in the change set with version 2.

Figure 25 Compare with Change Set Predecessor



To Compare with a Change-Set Predecessor

1 Display an activity properties sheet.

   a In ClearCase Explorer, select your work area and click **My Activities**.

   b Right-click the activity and click **Properties**.

2 In the activity properties sheet, click **Change Set**.

3 Right-click the version you want to compare, and, on the shortcut menu, click **Compare with Change Set Predecessor**.

Tracking Checked-Out Versions

Depending on how you work, you may forget exactly how many and which files are checked out. To list all the files and directories you currently have checked out to your view:

1. In the ClearCase Explorer Folder window, right-click a folder for a versioned directory and click **Find Checkouts**.
2. In the Find Criteria dialog box, in **Search folder**, edit the path from the root directory of the snapshot view.
3. Under **Include additional folders?**, select **Include subfolders**.
4. Click **OK**.

Viewing the Change Set of an Activity

You can see which versions are in an activity change set from ClearCase Explorer or Project Explorer.

To View the Change Set of an Activity

Do either of the following:

- From the view in ClearCase Explorer:
  a. From MyActivities in ClearCase Explorer, right-click an activity.
  b. On the shortcut menu, click **Properties**.
- From the ClearQuest user database:
  a. In the Workspace pane, click **Public Queries > UCMUserQueries**.
  b. Right-click MyToDoList and click **Run**.
  c. The list appears in the Query Builder pane.
  d. In the Query builder pane, select a record.
  e. In the Record form pane, display the change set list. Your project manager determines the location of this list when she designs the record type.

To View the Change Set of an Activity from Project Explorer

1. In the ClearCase Explorer Shortcut pane, click **Toolbox**. Then click **UCM > Project Explorer**.
2. In Project Explorer, select a stream and right-click an activity in the stream; on the shortcut menu, click **Properties**.
3. In the Properties dialog box, click **Change Set**.

Moving Versions to a Different Activity

You can move versions from a ClearCase properties sheet, or, if your project is enabled for ClearQuest, from the ClearQuest user database.

To Move Versions Within a ClearCase Context

1. Display an activity change set. (For more information, see *Viewing the Change Set of an Activity* on page 85.)
2. On the **Change Set** tab, select one or more elements.
3. Right-click a selected element and click **Move to Activity**.

To Move Versions Within a ClearQuest User Database Context

1. Display an activity change set in the ClearQuest user database. For more information, see *Viewing the Change Set of an Activity* on page 85.
2. Select one or more elements from the change set list.
3. Right-click a selected element and click **Move to Activity**.
4. Follow the ClearCase prompts.

Canceling Checkouts

If you check out a file but do not want to check in your changes or want to start with a fresh copy, you can cancel the checkout as follows:

1. In the ClearCase Explorer Details pane, select one or more checkouts. Then right-click.
2. On the shortcut menu, click **Undo checkout**.
3. Click **Yes** in the Confirm Undo Checkout dialog box. You can choose to save any of your changes in the file *filename*.keep.

Under the Hood: Canceling Checkouts

When you cancel the checkout of a file element, ClearCase handles the request as follows:

1. It prompts you to rename the file in your view to *filename*.keep.
2. It notifies the VOB that you no longer have the version checked out in your view.

3 In a snapshot view, it copies from the VOB the version that was in your view when you performed the checkout operation.

In a dynamic view, it uses the config spec version-selection rules to select a version.

Canceling Directory Checkouts

Although you rarely need to check out a directory explicitly, if you do and then cancel the checkout, ClearCase notifies the VOB that you no longer have the version of the directory checked out to your view. ClearCase does not prompt you to rename a canceled directory checkout to *directory-name*.keep.

If you cancel a directory checkout after changing its contents, any changes you made with **cleartool rmname**, **mv**, and **ln** are lost. Any new elements you created with **mkelem** or **mkdir** become orphaned. (When you create elements from the graphic user interface with the **Add to source control** command, ClearCase checks out the directory, adds the element, and checks in the directory, avoiding the creation of orphaned elements.) ClearCase moves orphaned elements (and any data that exists in the view at the path of the new element) to the VOB lost+found directory under names of the form: *element-name.UUID*.

In such cases, **uncheckout** displays this message:

```
cleartool: Warning: Object "prog.c" no longer referenced.
cleartool: Warning: Moving object to vob lost+found directory as
"prog.c.5f6815a0a2ce11cca54708006906af65".
```

In a snapshot view, ClearCase does not remove *view-private objects* or start the update operation for the directory in the view. To return the directory in your view to its state before you checked it out, you must start the Update Tool. For information about starting the Update Tool, see Help.

In a dynamic view, ClearCase does not remove view-private objects, but it does revert the view to its previous state.

To move an element from the lost+found directory to another directory within the VOB, use the **cleartool mv** command.

To permanently delete an element in the lost+found directory, take note of the name of the orphaned element and use this command:

cleartool rmelem *VOB-path*\lost+found*orphaned-element-name*

For example, from a dynamic view:

cleartool rmelem \guivob\lost+found\prog.c.5f6815a0a2ce11cca54708006906af65

From a snapshot view:

```
cd c:\pat_v1.4_cropcircle_sv
cleartool rmelem guivob\lost+found\prog.c.5f6815a0a2ce11cca54708006906af65
```

Note: In a snapshot view, ClearCase treats the lost+found directory, which is located immediately below the root directory of a VOB, as any other directory. To load the directory in your view, you must use a load rule that specifies either the parent directory of the element or the directory itself. However, as with any other directory in a snapshot view, you do not need to load the lost+found directory to issue ClearCase commands for elements in the directory.

Canceling the Checkout of a Deleted File

If you check out a file element in a dynamic view and then delete that file from your view (by using, for example, the **Delete** command in ClearCase Explorer), you should cancel the checkout.

1. In ClearCase Explorer, right-click the directory containing the deleted file and click **Find Checkouts**.
2. Complete the Find Criteria dialog box.
3. In the Find Checkouts window, select the file whose checkout you want to cancel and click **Undo Checkout**.

In a snapshot view, deleting checked-out elements executes an update operation on the affected elements, after which you cannot cancel the checkout.

Checking In Files

Check in a file when you want to keep a record of the file in its current state. When you check in from your development view, ClearCase records the new version in your development stream; even though the new version is a permanent part of the element, the modifications are unavailable to the project team until you deliver them.

To Check In Files

1. In ClearCase Explorer, select one or more files.
2. Right-click a selected file and click **Check In**.
3. In the Check In dialog box, ClearCase displays the comments you entered when you checked out the file. You can reuse these comments or modify them.

Try to enter meaningful comments. Comments are a valuable part of keeping track of your work. You can search for text in comments to identify a specific version.

Snapshot View: Checking In VOB Links

When you issue a **checkin** command from a snapshot view, ClearCase handles the request as follows:

1. It copies your modifications to the VOB as a new version.

   The version you check in remains in the view, regardless of the view config spec.

2. It removes write permission for the file.

For any other instance of the file loaded into a snapshot view, ClearCase copies the new version from the VOB into your view. (If your load rules specify an element that appears in more than one VOB location, the element is copied into each of the appropriate locations in the directory tree of your view.)

When you check in a symbolic-linked directory from a snapshot view, ClearCase does not update any other instances of the directory loaded in your view. As you add file elements to source control, ClearCase adds a copy of the element to all instances of a parent directory loaded in your view.

Testing Your Work

To reduce the amount of merging needed when you deliver your work, you should build and test the modifications in your work area. If you work in a multiple-stream project, do this especially after you rebase the stream and before you deliver your work. If you work in a single-stream project, do this before you check in your changes.

If your organization uses **clearmake** or **omake**, you can use these ClearCase build tools for your test builds; however, the *build auditing* and *build avoidance* features are available only from dynamic views.

For more information, see *Building Software* and the **clearmake** or **omake** reference pages in the *Command Reference*.

Indicating Your Progress

If your project is enabled for ClearQuest, you can complete the following tasks to indicate your progress on an activity:

- Modifying information in ClearQuest
- Closing an activity
- Using schema-specific actions
- Reassigning an activity
- Deleting an activity

You may need to reassign or delete an activity whether or not your project is enabled for ClearQuest.

Modifying Information in a ClearQuest User Database

Your ClearQuest activities may include fields for keeping additional information. For example, an activity record form may contain an **Attachments** tab to link to supporting documents written in other applications. The form may also contain fields to describe the symptoms of a defect. Adding or modifying such information does not change the activity state or have any effect on the corresponding activity in your stream.

For information about adding information to multiple activities, see Help for ClearQuest.

To Modify ClearQuest User Database Information

1. Use a query to find the activity. For example, to use MyToDoList:
   a. Start the ClearQuest client (see *To Start the ClearQuest Client* on page 63).
   b. In the left pane of the client window, click **Workspace > Public Queries > UCMUserQueries** and run the MyToDoList query.

      ClearQuest displays MyToDoList in the Query builder pane. (Figure 19 on page 64.)
2. In the Query builder pane, select a record.
3. In the Record form pane, click **Actions > Modify**.

   Your project manager can include help for the record form. To see information about a field, right-click it and then click **Help**.
4. In the Record form, make your changes. (Changing the Owner field reassigns the activity. For information about reassigning an activity, see *Reassigning Activities* on page 92.)
5. To save your changes, click **Apply**. To clear changes you made without saving them, click **Revert**.

Closing an Activity

If you work in a multiple-stream project, your project manager may have set up a policy for your project that closes activities in ClearQuest when you deliver them.

If these policies are not in effect for your project, close your activity when you have completed the associated work (which is usually when you deliver it). Completed activities appear in MyCompletedWork, but they do not appear on MyToDoList.

However, completed activities remain in your stream and appear in your stream properties even after you deliver them. For information on queries, see *Queries, Reports, and Charts* on page 37.

If you work in a single-stream project that is enabled for ClearQuest, your project manager can set policies to have you perform other actions to finish activities (see *Finishing Your Activity* on page 97).

To Close a ClearQuest Activity

Your project manager may have customized your ClearQuest user database schema to implement various ways to move an activity to a state based on the Complete-state type. The following procedure assumes that your database schema includes an action to close activities:

1. Use a query to find the activity. For example, to use MyToDoList:
   a. Start the ClearQuest client (see *To Start the ClearQuest Client* on page 63).
   b. In the left pane, click **Workspace > Public Queries > UCMUserQueries** and run the **MyToDoList** query.

      ClearQuest displays MyToDoList in the Query builder pane. (Figure 19 on page 64.)
2. In the Query builder pane, select a record.
3. In the Record form pane, click **Actions** and select an action that closes activities.

   Your project manager can include help for the record form. To see information about a field, right-click it and then click **Help**.
4. Click **Apply**.

Using Schema-Specific Actions

Your ClearQuest user database schema, which your project manager can customize (see *The UCM-ClearQuest Schema* on page 34), may support these common change-request management actions:

- Marking activities as duplicate
- Reopening closed activities
- Postponing activities

Your database schema may provide various ways to use these actions. For example, record types that can be postponed may have a **Postpone** command that you can use after clicking **Actions** in the Record form pane of the record.

Keep the following UCM-ClearQuest integration concepts in mind while using any schema-specific actions:

- Your project manager can set up each UCM-enabled record type to behave differently; some actions may not be available for all record types.
- You cannot move an activity from one development stream to another. However, you can move versions in an activity change set to another activity in the same stream (see *To Move Versions Within a ClearCase Context* on page 86) or to an activity in a different stream (see *Moving Work from a Development Stream* on page 141).
- You cannot delete an activity that is either currently set in a view or that contains versions in its change set. The UCM View field in the ClearQuest Query results pane indicates whether an activity is currently set. (Figure 19 on page 64.) The **Change Set** list, which is available from the ClearQuest Record form pane or from the activity properties sheet, displays the activity change set.
- We recommend against working in a development stream that contains activities you do not intend to deliver. Keeping undelivered activities in your development stream increases the risk that, after a rebase operation, your development stream does not contain a cohesive set of versions. If you begin working on an activity and then postpone it until the next release, we recommend that you move the work to the new project, and then create and work in a new development stream. (For more information, see *Moving Work from a Development Stream* on page 141.)

Reassigning Activities

You can assign an activity on MyToDoList to another project team member by changing the activity owner, unless your project manager has enabled the UCM policy Perform ClearQuest Action Before Work On, which prevents you from assigning your work to someone else.

If the ClearQuest activity has been linked with a corresponding activity in your stream (which occurs when you set your view to the activity), you can change the owner in ClearQuest, but you cannot move the UCM activity to the new owner's development stream. The UCM Stream field in the ClearQuest Query results pane indicates whether an activity is linked with a stream. (Figure 19 on page 64.)

To reassign an activity that is linked with a stream, change the activity owner in ClearQuest, and then do one of the following:

- If you have not created versions for the activity (that is, the activity change set is empty), you can remove the activity in your development stream. (See *To Delete a UCM Activity* on page 93.) Then, when the new activity owner sets his view to the activity, ClearCase creates a new activity in the new owner's stream. The activity

Change Set list, which is available from the ClearQuest Record form pane or from the properties sheet of the activity, displays its change set.

- If the activity change set is not empty, the new owner can create and work in a view attached to the stream containing the activity. For information about attaching a new view to an existing stream, see Help.

To Change an Activity Owner

1. Use a query to find the activity. For example, to use **MyToDoList**:
   a. Start the ClearQuest client (see *To Start the ClearQuest Client* on page 63).
   b. In the left pane of the client window, click **Workspace > Public Queries > UCMUserQueries** and run the **MyToDoList** query.

      ClearQuest displays MyToDoList in the Query builder pane. (Figure 19 on page 64.)
2. In the Query builder pane, select an activity.
3. In the Record form pane, click **Actions > Modify**.
4. In the **Owner** field, select the new owner.
5. Click **Apply**.

Deleting Activities

Note: Your project manager determines whether you can delete ClearQuest records of any specific record type.

If a ClearQuest activity is linked with a UCM activity (which happens if you set a view to the activity) and if the UCM activity contains versions in its change set, you must delete the UCM activity first.

The UCM Stream field in the ClearQuest Query results pane indicates whether an activity is linked with a stream. (Figure 19 on page 64.) The activity **Change Set** list, which is available from the ClearQuest Record form pane or from the properties sheet of the activity, displays its change set.

To Delete a UCM Activity

Note: You cannot delete an activity if the change set contains one or more versions. We recommend that you use some other action for an activity with a nonempty change set. For example, you can deliver all other activities in the stream, abandon the stream, and create a new one to keep track of your work on the project. But if no other action is appropriate, you must first use **cleartool rmver** to remove the

versions from the VOB. Removing versions from the VOB is irreversible and may have unintended consequences, for example, if there are dependencies on the versions that you remove.

1 In ClearCase Explorer, from the Shortcut pane, click **Toolbox > UCM > Project Explorer**.

2 In Project Explorer, navigate to your project.

3 Select the stream that contains the activity.

4 In the right pane, select the activity and click **File > Delete**. If the activity is linked with a record in ClearQuest, you must delete the ClearQuest record from the ClearQuest graphic user interface.

To delete an activity from a command prompt, use **cleartool rmactivity**. For more information, see the **rmactivity** reference page in the *Command Reference*.

To Delete a ClearQuest Activity

1 Use a query to find the activity. For example, to use MyToDoList:

   a Start the ClearQuest client (see *To Start the ClearQuest Client* on page 63).

   b In the left pane of the client window, click **Workspace > Public Queries > UCMUserQueries** and run the MyToDoList query.

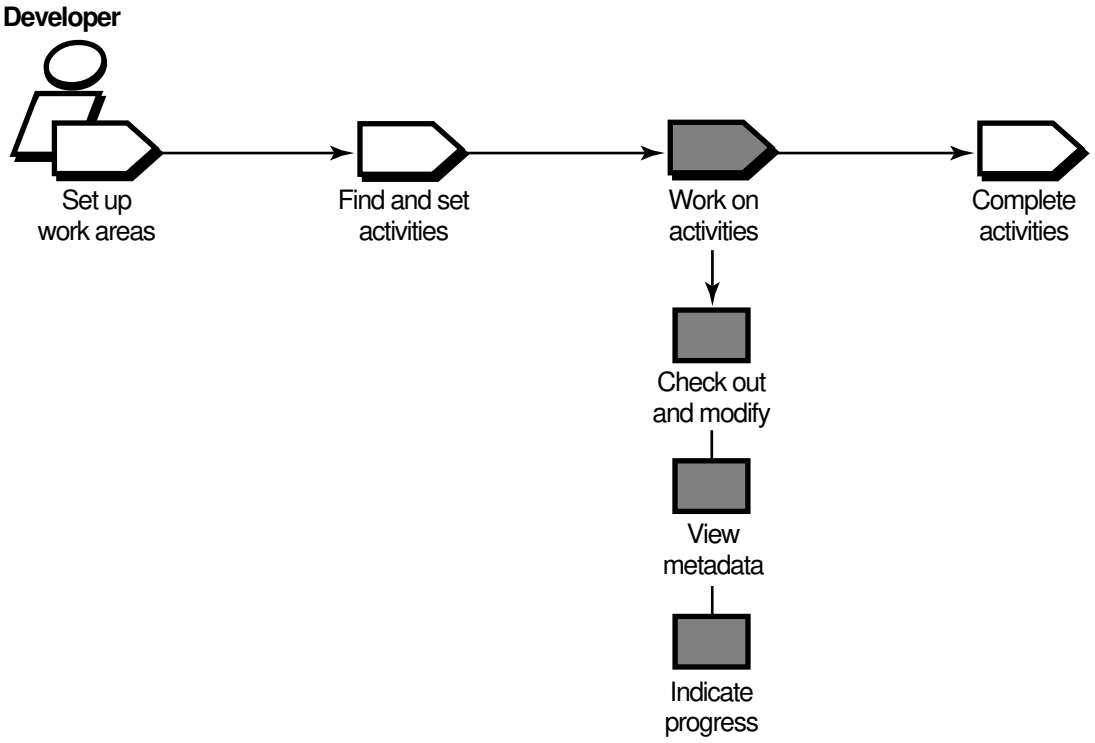
   ClearQuest displays the records in the **Result set** tab in the Query builder pane. (Figure 19 on page 64.)

2 In the Query builder pane, select an activity.

3 In the Record form pane, click **Actions > Delete**. If a ClearQuest activity is linked with a UCM activity (and the activity change set is empty), ClearQuest deletes the activity in the stream as well.

4 Click **Apply**.

Working in a Single-Stream Project



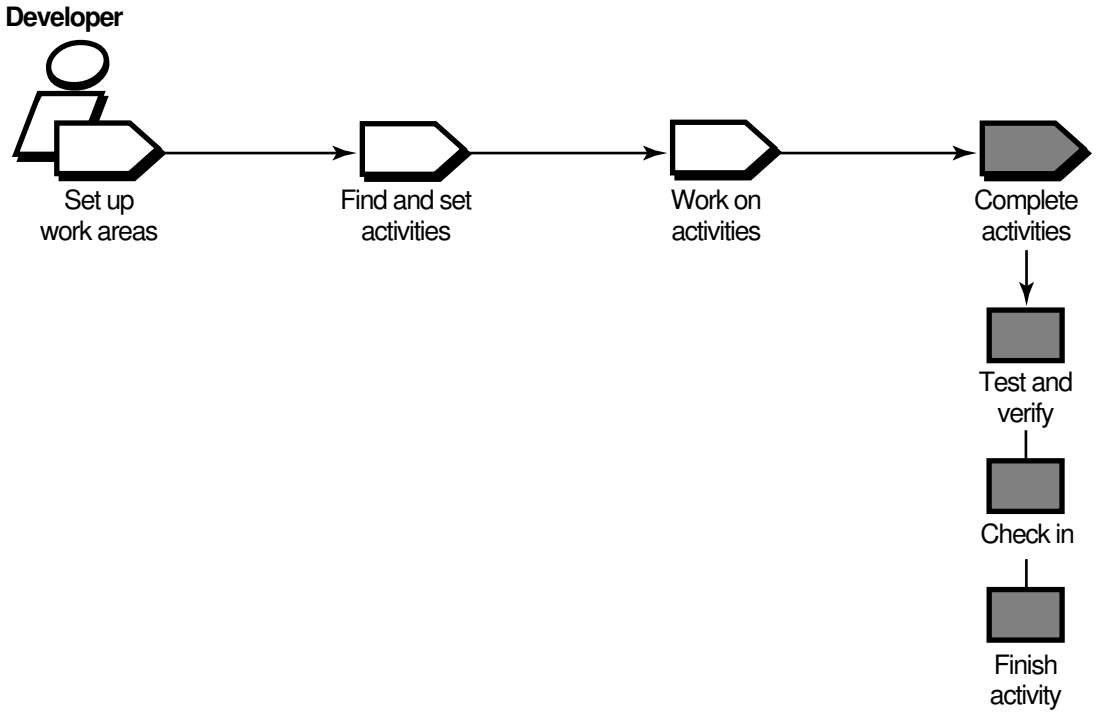
Working on activities in single-stream project involves the following tasks:

- Checking out files (see *Checking Out Files* on page 76)
- Working with checkouts (see *Working with Checkouts* on page 83)
- Canceling checkouts (see *Canceling Checkouts* on page 86)
- Indicating your progress (see *Indicating Your Progress* on page 89)

Tasks While Working in a Single-Stream Project

The tasks for working in a single-stream project are similar to those one does in a multiple-stream project. However, you do not check in your work until you are ready to complete activities in the project (see *Completing Activities in a Single-Stream Project*).

Completing Activities in a Single-Stream Project



In a single-stream project, when you are ready to make work in the project integration stream available to the project team (for example, to contribute to official project builds), you complete your activity. Completing your activity involves the following tasks:

- Testing your work (see *Testing Your Work* on page 89)
- Checking in files (see *Checking In Files* on page 88)
- Finishing your activity (if your project is enabled for ClearQuest)

In a single-stream project, you start work from the latest version in your integration stream, which you access through your integration view. When you check in your work, those versions are the latest versions in the integration stream. Team members working in dynamic views see your work immediately. Team members working in snapshot views see your work after the update their views.

Because the work is available to others on the project when you check in, you must test your work *before* you check in. Because checking in makes your work available to others, there is no need to deliver or rebase.

Also, if your project is enabled for ClearQuest, you must finish activities (see *Finishing Your Activity* on page 97).

Finishing Your Activity

In a single-stream project that is enabled for ClearQuest, you must indicate the completion of your work by finishing the activity. If your project uses the UCM-ClearQuest integration, your project manager may enable a policy that runs a customized ClearQuest user database script when you finish an activity. For example, your project manager may create a script that checks whether your user name matches the name in the ClearQuest user database record Owner field or asks you to enter other information. When you successfully complete the script, another policy can automatically transition your activity to the complete state in the ClearQuest user database.

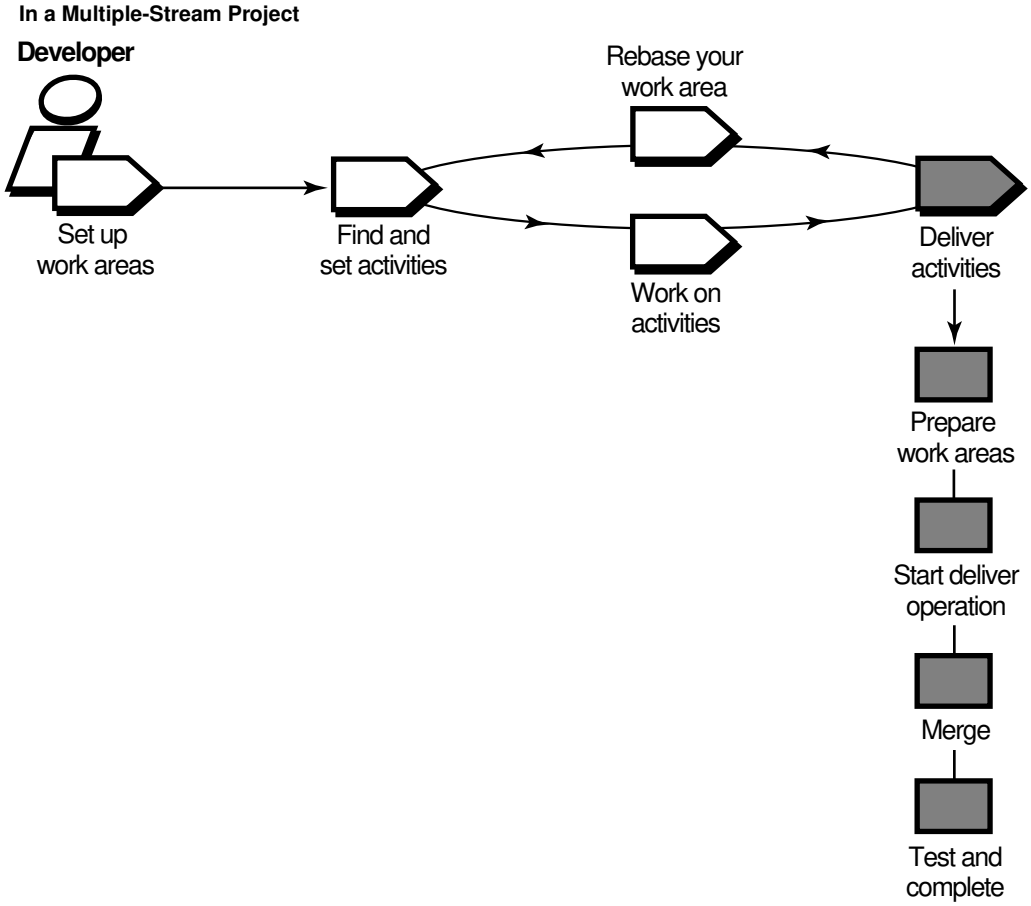
To finish your activity, do the following:

1. In ClearCase Explorer, find your activity (see *Finding Activities with the My Activities Folder* on page 62).
2. In the Details pane, right-click the activity and click **Finish Activity**.

If any versions in the activity are checked out, the Check Out dialog box appears for you to check in your changes. After you check in, this transitions the activity to the Complete state type (see *State Types and State Transitions in a Schema Enabled for UCM* on page 42).

Delivering Activities

5



When you are ready to make work in your development stream available to the project team (for example, to contribute to official project builds), you deliver your work. Delivering your work involves the following tasks:

- Preparing your work areas
- Starting the deliver operation
- Selecting activities and baselines

- Merging versions
- Testing your work
- Completing the deliver operation

  Note: If your project uses Rational ClearCase MultiSite to share source data with developers in other geographical locations and if the project streams are mastered at a different site, you complete only the first three tasks; the project integrator completes the remaining tasks. For more information, see *MultiSite: Posting Work to Deliver* on page 117.

Preparing Your Work Areas

Before you start the deliver operation, prepare your work areas as follows:

- If your project integrator has created a new *recommended baseline* since you last rebased, rebase your development work area to minimize the amount of merging needed during the deliver operation. For information about rebasing, see Chapter 6, *Rebasing Your Work Area*.
- In your development work area, find, compare, and check in all the work you want to deliver.
- If your integration view is a snapshot view, update it and resolve any *hijacked files*.

Finding, Comparing, and Checking In Work from Your Development View

Rational ClearCase delivers only checked-in changes. In addition, your project manager may have set a project policy that requires you to check in all files and directories in your development view before starting the deliver operation.

To Find, Compare, and Check In Your Work

1. In the ClearCase Explorer Folder pane, right-click a folder for a versioned directory and then click **Find Checkouts**.
2. In the Find Criteria dialog box, select these criteria:
   - Under **Include additional folders?**, select **Include subfolders**.
   - Under **Search only for your checkouts**, select **Yes**.
   - Under **Search only for checkouts in the current view**, select **Yes**.
3. Then click **OK**.

   Find Checkouts produces a list of file or directory versions checked out to your development view and indicates which activity each checked-out version belongs to.

4 To see which changes are in a checked-out version:

- To see the differences between the previously checked-in version and the current version, click **Tools > Compare with Previous Version**.
- To see checkin comments, click **Tools > History**.

5 Do **one** of the following:

- To deliver changes for a specific file, check it in by selecting it and clicking **Tools > Check In**.
- To undo your changes for a file, cancel its checkout by selecting it and clicking **Tools > Undo Checkout**.

  During the cancel checkout operation, you can choose to save your changes in a *view-private file* (see *Canceling Checkouts* on page 86).

6 To keep your changes without delivering a file, leave it checked out.

If you leave a file checked out, ClearCase delivers the predecessor version in the activity change set. If the checked-out version is the only version in the change set, ClearCase does not deliver any version of the element.

Updating Your Integration View

During the deliver operation, use your integration view to make sure that your delivered work builds successfully with the work other team members have delivered to the target stream.

If your integration view is a snapshot view, update it and resolve any *hijacked files* to ensure that it contains the latest versions delivered to the target stream.

To Update a Snapshot View and Resolve Hijacked Files

1 In ClearCase Explorer, right-click the root directory of the view and click **Update View**.

2 In the Start Update dialog box, click **Advanced** and select an option for resolving *hijacked files*. (For information about each option, see the Help.)

3 To see the actions needed to update the view without actually updating, click **Preview only** on the **View** tab.

4 Click **OK**.

ClearCase displays its progress as it updates the view (or previews the update).

5 In the Update window, check out or undo any hijacked files. For more information, see *Using the Update Tool* on page 155.

Starting the Deliver Operation

In UCM, you can deliver your work as activities or as baselines. Typically, in a single project environment, you work on activities and deliver your completed activities. Some UCM environments manage parallel work on the same components. The project integrators in such environments deliver baselines to the project integration stream to incorporate changes in components. For information about delivering baselines, see *Managing Software Projects*. This section covers delivering activities.

You can deliver your work to either the default target (the parent stream of your development stream) or a nondefault target. The policies on the project and on the target stream determine what target is allowed. When you start the deliver operation, the target stream policy settings are checked to determine whether the deliver operation is allowed. The project policy determines the settings for streams in the project; the project policy may defer the setting to the target stream policy. For information about project and stream policies, see *Managing Software Projects*.

If your organization uses ClearCase MultiSite and your site does not master the target of your delivery, you can only post your work (see *MultiSite: Posting Work to Deliver* on page 117).

Deliveries and the UCM-ClearQuest Integration

If your project uses the UCM-ClearQuest integration, your project manager may enable a policy called Perform ClearQuest Action Before Delivery. This policy runs a customized ClearQuest user database script when you start the deliver operation. For example, your project manager may create a script that checks for dependencies between ClearQuest user database records. Before you can continue the deliver operation, you must satisfy any requirements established by the Perform ClearQuest Action Before Delivery policy.

If your organization uses ClearCase MultiSite and the UCM-ClearQuest integration, the activity object and its ClearQuest user database record must be mastered locally before you can work on, set, or change an activity.

Delivering to the Default Target

If you deliver your work to the default target, the target of the delivery is your parent stream (see *Your Parent Stream* on page 21).

To Start the Deliver Operation to the Default Target

You can start the delivery from either ClearCase Explorer or Windows Explorer.

From ClearCase Explorer

Do **one** of the following:

- Use the Toolbox.
  - **a** In the ClearCase Explorer Shortcut pane, click **Toolbox**. Then click **UCM > Deliver from Stream**.
  - **b** Select your development stream and click **OK**.
- Right-click the root directory of your development view and select **Deliver from Stream to Default**. (For more information, see *Accessing Your Development View* on page 54.)

From Windows Explorer

Right-click the root directory of your development view and then click **ClearCase > Deliver from Stream to Default**.

When the deliver operation starts, you have to select what to deliver (see *Selecting Activities* on page 104).

Delivering to a Nondefault Target

If you are delivering your work to a nondefault target, you can choose a stream in the same project (intraproject delivery) or a stream in another project (interproject delivery). The policies on the target stream (intraproject) or on the target project (if you are doing an interproject delivery) determine what target is allowed. You can attempt to deliver from any stream in a project to any other stream in the same project, including the integration stream. Whether the operation can proceed depends on the applicable policy settings. See *Managing Software Projects* for details on policies.

To Start the Deliver Operation to a Nondefault Target

You can start the delivery to a nondefault target from either ClearCase Explorer or Windows Explorer.

From ClearCase Explorer

1. In the ClearCase Explorer, access your view and right-click the root directory of your development view. (For more information, see *Accessing Your Development View* on page 54.)
2. Then click **Deliver from Stream to Alternate Target**.

3 In the Deliver from Stream dialog box, select the target stream and click **OK**.

From Windows Explorer

1 Right-click the root directory of your development view and then click **ClearCase > Deliver from Stream to Alternate Target**.

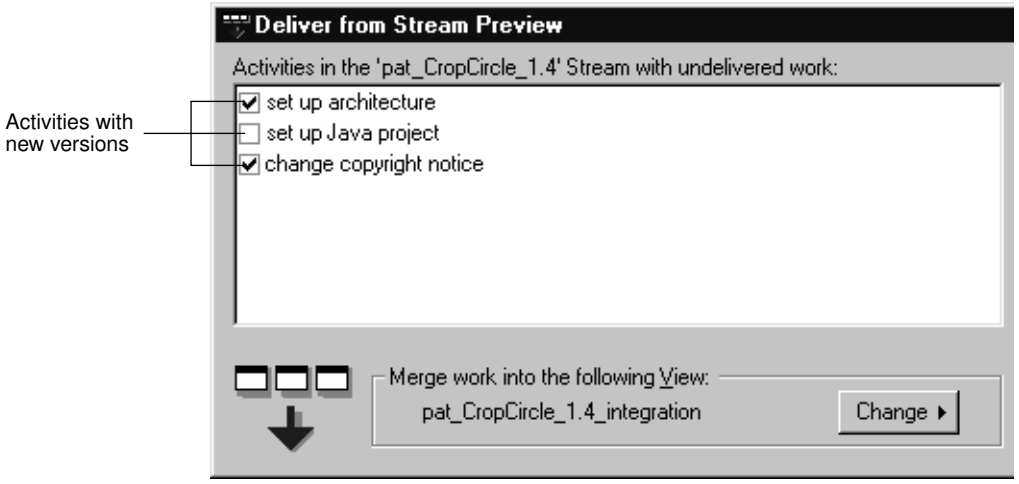
2 In the Deliver from Stream dialog box, select the target stream and click **OK**.

The Deliver from Stream Preview dialog box opens. Now you select activities to deliver (see *Selecting Activities* on page 104).

Selecting Activities

After you start the deliver operation and, optionally, specify the target, ClearCase opens the Deliver from Stream Preview dialog box to display the activities in your development stream that have never been delivered or that include versions created since the last deliver operation (Figure 26).

Figure 26 Undelivered Activities



The Delivering to Target box shows the target context of your delivery.

You can select all activities in the dialog box or a subset. If you select a subset of the activities, ClearCase checks for dependencies between versions in the change sets before continuing the operation. If your selection does not meet dependency requirements (see *Activity Dependencies in the Deliver Operation*) and baseline considerations (see *Baseline Considerations in the Deliver Operation* on page 106), ClearCase prompts you to change your selection.

To see an activity change set, select the activity, click **Properties**, and click the **Change Set** tab in the Properties dialog box.

Controlling Merge Behavior

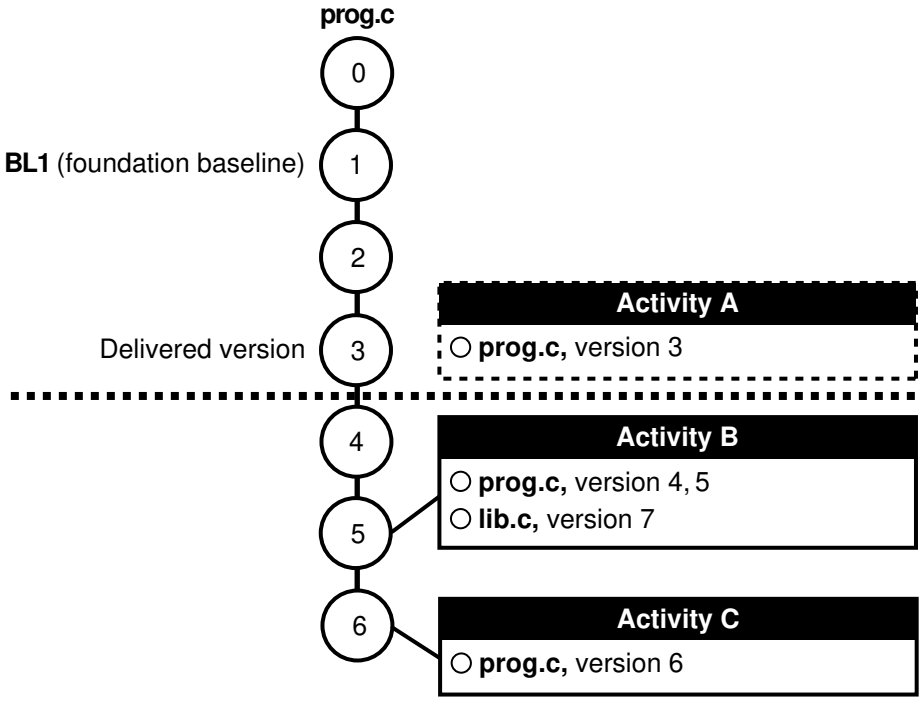
ClearCase tries to do the merge automatically. However, you can perform the merge on each element manually. To handle such a need, in the Deliver from Stream Preview dialog box, select **Merge elements graphically** to turn off all automated merging. If you select this option, the merge operation prompts you for confirmation before proceeding with each change.

Activity Dependencies in the Deliver Operation

For any given element in an activity change set, you must deliver all versions in your stream in the range between the version in the change set and either the version most recently delivered or the version in the stream *foundation baseline* (whichever is later). For example, in Figure 27, because version 3 of prog.c was delivered in a previous deliver operation, the dependencies between Activities B and C, which contain versions of prog.c, are as follows:

- You can deliver Activity B without delivering Activity C.
- To deliver Activity C, which contains version 6, you must also deliver Activity B, which contains the versions between 6 and the most recent delivery of prog.c.
- Because Activity B also contains versions of lib.c, you may be required to deliver other activities to satisfy dependencies for lib.c. For example, if Activity D (not shown in Figure 27) contained an undelivered version 6 of lib.c, delivering Activity B would require you to deliver Activity D as well.

Figure 27 Activity Dependencies

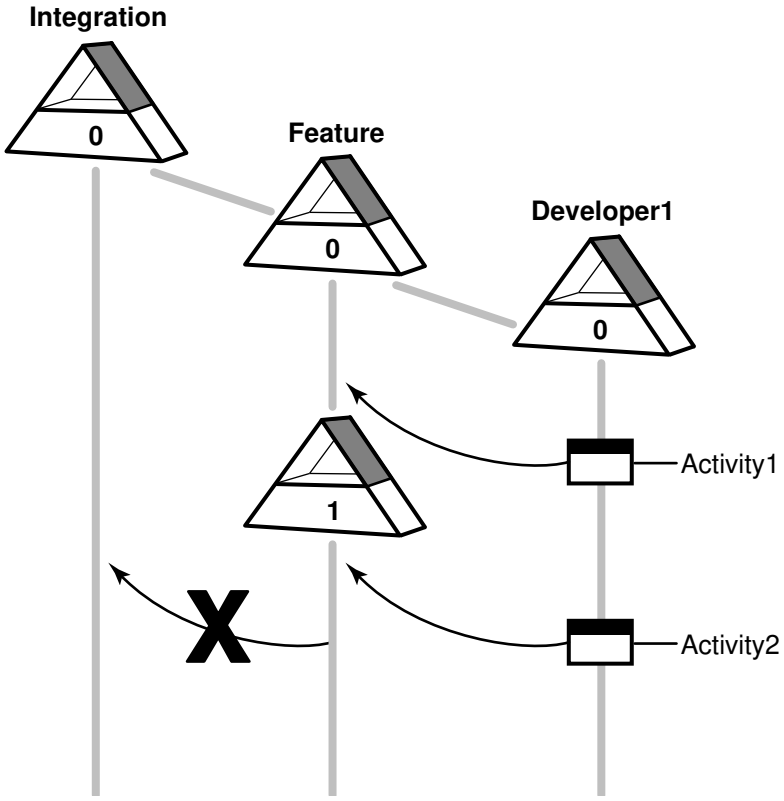


Baseline Considerations in the Deliver Operation

As with dependencies between activities in the deliver operation (see *Activity Dependencies in the Deliver Operation* on page 105), in some situations, baselines can affect the deliver operation. If the foundation baseline of the source stream is different than that of the target stream, the deliver operation includes activities in the foundation baseline plus all activities created after the foundation baseline. This prevents you from delivering only selected changes.

For example, the Feature stream in Figure 28 is configured with baseline 0. The developer working in the Developer1 stream delivers Activity1 to the Feature stream.

Figure 28 Baseline Consideration in Deliver

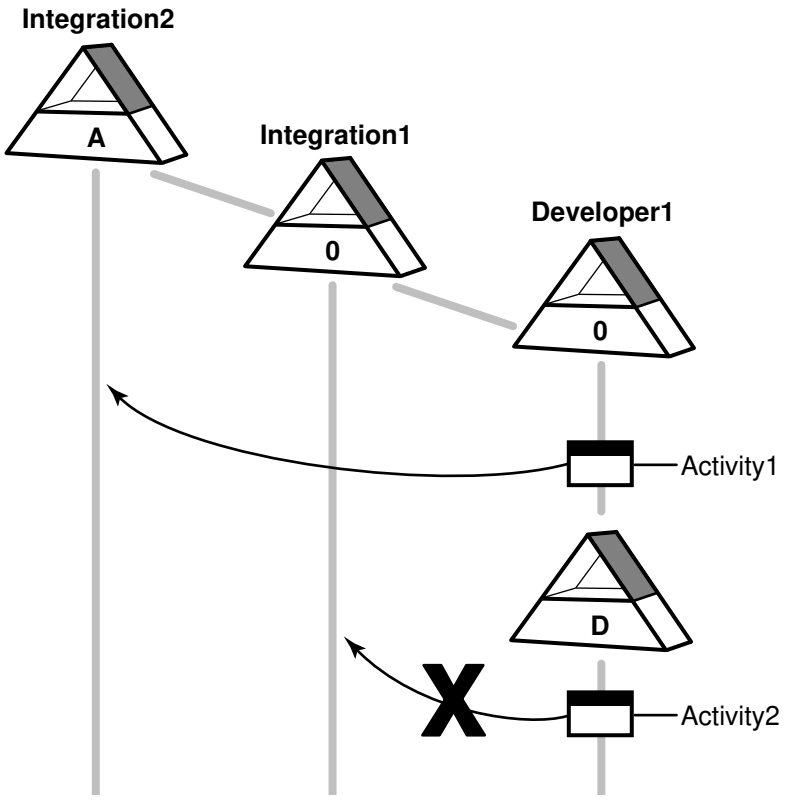


The owner of the Feature stream explicitly creates baseline 1 in that stream. Activity Activity1 is included in baseline 1. The developer in the Developer1 stream continues working on a new activity and delivers Activity2 to the Feature stream.

If the integrator of the Feature stream delivers work to the Integration stream, the delivery includes activity Activity2 plus the activities included in baseline 1. The integrator cannot deliver only Activity2.

Another baseline consideration occurs when you deliver to an alternative target stream, as shown in Figure 29.

Figure 29 Baseline Consideration in Alternative Target Deliver



The Developer1 stream in one project delivers Activity1 to an alternative target, the integration stream (Integration2) in another project. As an intermediate step, deliver creates a hidden baseline D (referred to as a deliver baseline) in the Developer1 stream. If the Developer1 stream next tries to deliver only Activity2 to the integration stream (Integration1) in its project, deliver requires that the changes associated with the deliver baseline (Activity1) be delivered. In this case, you cannot deliver only activity Activity2, but must also deliver activity Activity1 which is included in the deliver baseline.

Merging Versions

Note: If your project uses Rational ClearCase MultiSite to share source data with developers in other geographical locations, and if the project streams are mastered at a different site, you do not merge versions. For more information, see *MultiSite: Posting Work to Deliver* on page 117.

After selecting activities, click **OK** in the Deliver from Stream Preview dialog box to start merging versions in your development view with versions in the integration view. As it merges files and directories, ClearCase displays its progress in the Delivering to View dialog box.

For each new version delivered to the integration view, ClearCase merges all nonconflicting differences.

As it merges versions, ClearCase changes the value in the Merged column of the Delivering to View dialog box to **Yes**.

If versions in your development view contain changes that conflict with the corresponding versions in the integration view, the value in the Merged column shows **No**. If the file is a text file, ClearCase starts Diff Merge, a tool that helps you resolve conflicting differences. If the file is an XML type, XML Diff Merge starts.

Under the Hood: Concurrent Deliver Operations

ClearCase allows developers to deliver work concurrently. However, if another team member's deliver operation has checked out an element, you cannot deliver any changes to that element until that deliver operation is completed or canceled.

When delivering activities, ClearCase follows a specific order:

1. It evaluates each directory sequentially, checking out and merging changes to the target stream when needed. If it encounters a reserved directory checkout in the target stream, it stops the deliver operation (but does not undo any successful directory merges).
2. It then attempts to check out each element that requires a merge. If it encounters an element that it cannot check out (for example because another team member's deliver operation has checked out the element), it skips the element and attempts to check out the remaining elements.

   If it is unable to check out an element, it does not proceed to the following step. In this case, you have two options:

   - Wait until the other developer completes or cancels her deliver operation, and then restart your deliver operation. Note, however, that with a deliver operation in progress, any new versions you create in your development stream are not delivered until the next deliver operation.
   - Undo your deliver operation and continue working on your activities. For more information, see *To Undo a Deliver Operation* on page 115.
3. After successfully checking out all elements, it merges each element, starting Diff Merge and requesting your input when it encounters merge conflicts.

Under the Hood: Integration Activities and Baselines

ClearCase uses a special kind of activity, an integration activity, to keep track of the changes that occur during deliver and rebase operations. ClearCase creates the integration activity and adds versions to it; you do not need to create or maintain it. You may want to view the properties of an integration activity to see exactly which versions were merged and created during a deliver or rebase operation. You can view integration activity properties from the Project Explorer; the activity name is of the form *deliver.stream-name.date-stamp*.

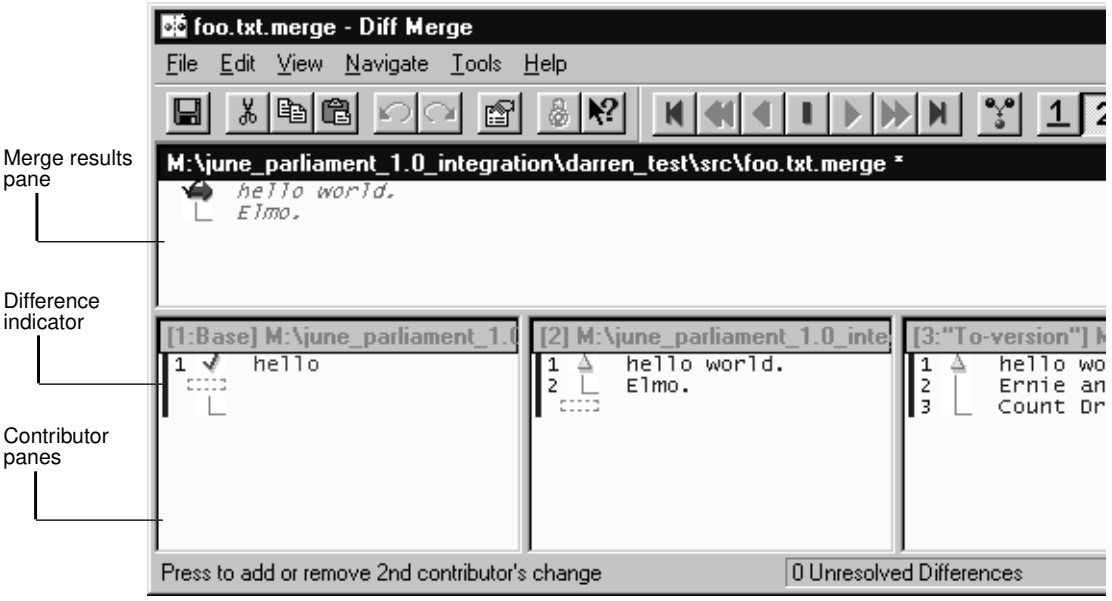
If your project uses the UCM-ClearQuest integration, ClearCase links a UCM-enabled record based on the UCMUtilityActivity record type with the UCM activity. During the deliver operation or the rebase operation, the activity is in a state based on the Active state type. After the operation, ClearCase moves the activity to a Complete state type.

As an intermediate step in the delivery, ClearCase creates a baseline in your development stream that identifies the state of your stream at the time of delivery. You can browse this baseline and its properties in the Component Browser. For more information, see *Managing Software Projects*.

Using Diff Merge to Resolve Differences

When Diff Merge starts, it indicates that it has resolved all nonconflicting differences. Then, it highlights the first unresolved difference with a vertical blue line at the left edge of the contributor panes (Figure 30).

Figure 30 Diff Merge Window



To Resolve Differences

1 In Diff Merge, compare the differences in the contributor panes and click the numbered button on the navigation toolbar that corresponds to the contributor pane number containing the resolution you prefer.

   For example, in Figure 30, clicking **2** moves the difference region in contributor pane 2 to the results pane.

   In some cases, you can click more than one numbered button to include changes from multiple contributor panes. In addition, after you select a contributor to resolve the difference, you can edit the results pane directly.

2 Click **Navigate > Next Unresolved Difference**.

3 Repeat Step 1 and Step 2 until you have resolved all unresolved differences.

4 Click **File > Save** and then click **File > Exit**. When you complete the deliver operation, ClearCase checks in the merge results.

   Note: For more information about using Diff Merge, choose a command from the Diff Merge **Help** menu.

After you exit Diff Merge, ClearCase continues merging the versions in your activities.

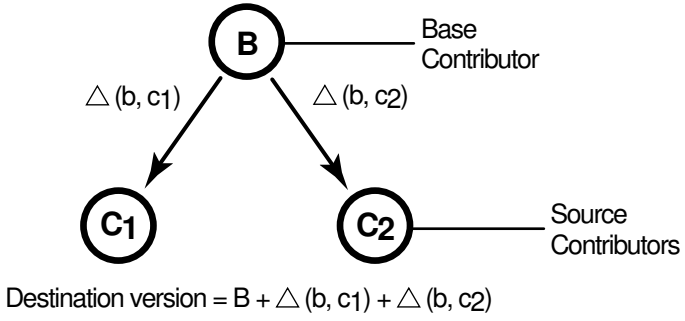
Under the Hood: How ClearCase Merges Files and Directories

A merge combines the contents of two or more versions into a single new version. During a deliver operation, the merge algorithm involves the following versions:

- The source contributors: one version from your development stream and one version from the target stream.
- The base contributor: the common ancestor of the source versions.
- The destination version: the merge output, which becomes a new version in the target stream.

As illustrated in Figure 31, ClearCase starts with the base contributor and adds changes that are present in the source contributors.

Figure 31 ClearCase Merge Algorithm



If the two source contributors contain conflicting changes (for example, if each contains a line that is different from the base contributor), ClearCase starts Diff Merge so you can choose which change to add to the destination version.

Handling a Binary File in a Deliver Operation

If you have a binary file in any of your activities when you perform the deliver operation, and your organization does not have a type manager for that file type, ClearCase generates an error when it encounters the binary file during the delivery. The error occurs because there is no type manager that can merge the version of the file in your development stream with the version of the file in the target stream. Your project manager can overcome this problem by creating a special element type for the binary file type and associating that element type with the binary file in the VOB. This special element type can tell the ClearCase system not to attempt merging the file. For more information, see *Managing Software Projects*.

To Stop and Resume the Deliver Operation

You can stop the deliver operation temporarily by clicking **Stop** in the Delivering to View dialog box.

To resume the deliver operation, click **Resume** or right-click the root directory of the view in ClearCase Explorer and click **Deliver from Stream**.

Delivering Changes to a Read-Only Component

Deliver cannot allow changes to a read-only component. This situation can result from someone changing the component modifiability in your project. What actions you take depend on whether you are doing an intraproject or an interproject deliver.

For Intraproject Deliver

If you are performing an intraproject deliver for a component that was previously modifiable, you receive a warning message. Decide what can be delivered from your stream. If the changes in the other components are independent of the changes in the read-only component, move the versions from the read-only component into a new activity. Deliver all activities in your stream except the new one.

If any of the changes in the other components are dependent on changes to the read-only component, decide whether those versions can be separated into a new activity. This may involve not delivering a subset of activities in your development stream.

If there are dependencies among the versions, your stream may no longer be deliverable to its default target. Decide whether there is a project to which you can deliver the changes. The project integrator may have to create a new development stream in this project and selectively find and merge the changes (findmerge -insert) from the versions into the new stream. Deliver cannot be used.

As a last resort for an intraproject deliver, you can remove all the versions (**rmver**) that should not have been made. Also, remove all changes dependent on that component. You must use **rmver -xhlink** to remove the versions from the change set. You may need to manually recode some changes to remove the dependence on the changes to the read-only component.

For Interproject Deliver

If you are performing an interproject deliver, you can also receive a warning message. The project from which you are delivering and the target project can have differences in component modifiability at any time. You have the following choices:

- Set the policies on the target stream and target project so that deliver can ignore the changes in both the read-only component and in other components that are dependent on the changes to the read-only component.
- Isolate the changes in the read-only component and changes in other components that are dependent on the changes to the read-only component. Deliver only the activities that contain changes to modifiable components.

Testing Your Work

At this point in the delivery, when all versions are merged, the deliver operation is still in progress. The destination versions (the new versions resulting from the merges) are checked out and visible only in your integration view.

Your integration view shows your delivered work and the work that other team members have delivered. To make sure that your work is compatible with other delivered work, we recommend that you test the work in your integration view before you complete the deliver operation.

To go to your integration view, click **OK** in the Merges Complete dialog box. ClearCase opens your integration view in ClearCase Explorer.

An additional dialog box, Deliver Status, remains open. Consider leaving it open to facilitate completing the deliver operation after you build and test.

Your organization can use **clearmake** or **omake** as ClearCase build tools; however, the *build auditing* and *build avoidance* features are available only from dynamic views. For more information about **clearmake** or **omake**, see *Building Software* and the reference pages in the *Command Reference*.

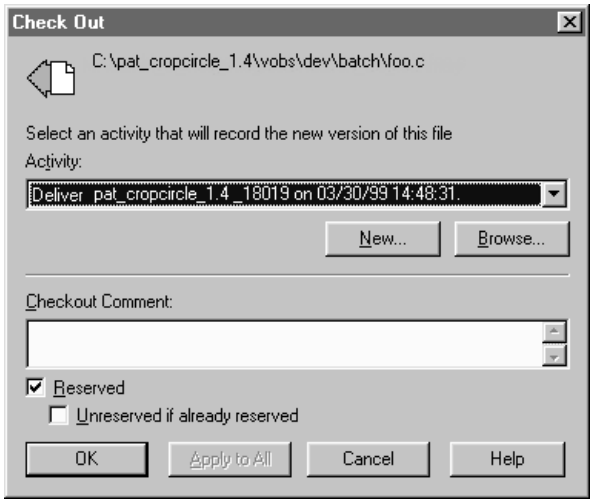
In addition to building and testing, you may need to do the following:

- Edit the versions checked out to your integration view to resolve build errors
- Check out and edit additional elements to resolve build errors
- Update your integration view
- Undo a deliver operation

Checking Out Versions During Testing

In the process of building and testing, you may need to check out versions in your integration view and modify them to resolve build errors. As described in *Under the Hood: Integration Activities and Baselines* on page 110, ClearCase adds any modifications you make during the deliver operation to an integration activity (Figure 32).

Figure 32 Integration Activity



Note: We recommend that you postpone checking in your modifications until you complete the delivery. Checking in complicates efforts to undo the deliver operation, if you find it necessary to do so.

Under the Hood: Checking Out Versions from Snapshot Views

If you and other project members perform concurrent deliveries and if your integration view is a snapshot view, the view becomes out of date when other team members deliver to the target stream. If you try to check out a version that is not the latest in the target stream, ClearCase prompts you to check out the latest version in the target stream and then loads it into your view.

Undoing a Deliver Operation

At any time before you complete the deliver operation, you can undo changes made as a result of the deliver operation. This undo procedure is valid only if you have started a deliver operation, have not checked in modifications in the target view, and have not completed the deliver operation.

To Undo a Deliver Operation

1. If you started Diff Merge, close it.
2. If the Deliver Status dialog box is open, click **Close**.
3. In ClearCase Explorer, right-click the view root directory and click **Deliver from Stream**.

4 In the dialog box, click **Undo the delivery** and click **OK**.

ClearCase cancels all checkouts in the integration view and removes all merge arrows created during the current deliver operation. (ClearCase uses merge arrows in VOBs to keep a history of merges.)

If you checked in any versions to the integration view, clicking **Cancel** is not sufficient to undo the deliver operation. You must either complete the operation (recommended) or use **cleartool rmver –xhlink** to remove the versions you checked in before canceling the operation. Because **rmver** erases part of your organizational development history, your organization may not allow individual developers to use it. Because it may have unintended consequences, be very conservative in using this command, especially with the –**xhlink** option. For more information, see the **rmver** reference page in the *Command Reference*.

Completing the Deliver Operation

Because the deliver operation is not instantaneous (for example, you may need to build, test, and correct errors several times during a deliver operation), ClearCase imposes a simple state model: deliveries are either *in progress* or *complete*, and you cannot start a new deliver operation from a development stream when one is already in progress for the stream.

Note: If a view in which a deliver operation is in progress has been deleted, you are prevented from completing (and resuming or canceling) the deliver. This can block all other deliveries for the project. To complete the deliver, use the **deliver** command as described in the *Command Reference*. For more information, see the **deliver** reference page.

To Complete a Deliver Operation

1 Open the Deliver Status dialog box as follows:

In ClearCase Explorer, right-click the view root directory and click **Deliver from Stream**.

2 In the dialog box, click **Complete the delivery** and click **OK**.

The Deliver Status dialog box can remain open during the deliver completion to report status as ClearCase attempts to place the deliver operation in the complete state.

What Happens When You Complete a Deliver Operation

When you issue the **complete** command for a deliver operation, ClearCase does the following:

1. It checks in the merge results in the target stream.
2. If your project is enabled for ClearQuest and your project manager has enabled a policy, other actions occur.
   - If the Perform ClearQuest Action After Delivery policy is enabled, a ClearQuest user database script executes. If the script returns an error, ClearCase stops the deliver operation. You must correct the error before the deliver can continue.
   - If the Transition to Complete After Delivery policy is enabled, the delivered activities move to a Complete-type state.
3. It changes the state of the operation to Complete.

MultiSite: Posting Work to Deliver

If your organization uses ClearCase MultiSite to distribute development among multiple geographical sites, your project may share source files with developers at other sites. Each site has its own replica of project *components*, and developers work in site-specific replicas. Each replica controls (masters) a set of streams, and only developers at that replica site can modify them.

If your site does not master a stream, you cannot complete deliver operations to that stream. UCM provides a variation of the deliver operation called a *remote deliver*. You must post your work to deliver. If the target stream is mastered by the local PVOB replica, the deliver operation is local.

After you deliver your work and the delivery is in the posted state, the project integrator at the site that masters the target stream completes the remote deliver operation.

To Deliver to a Nonmastered Target Stream

To make your work on a remote stream available in the target stream:

1. Prepare your work areas (see *Preparing Your Work Areas* on page 100).
2. Start the deliver operation (see *Starting the Deliver Operation* on page 102).

   In a remote deliver operation, the source stream must be a development stream.
3. Select activities (see *Selecting Activities* on page 104).
4. Deliver the work so that it can be posted.

The deliver operation determines whether your development and target streams are mastered at different replicas. If they are mastered at different replicas, a remote deliver operation is put into effect. ClearCase changes mastership of the development

stream object to match that of the target stream object, and it notifies you that it has assigned the development stream the posted status.

When your development stream has the posted status, notify the project integrator at the site that masters the target stream replica. The project integrator can then find posted deliver operations and either continue the operation or cancel it to return the development stream to its previous state. Note that, after you post your work, only someone at the master site can cancel or complete the operation.

You can create activities and perform checkins and checkouts for your development stream while the remote deliver is in process. However, you cannot add, remove, or create baselines; add or remove components; or rebase the development stream. The delivery completes when the posted deliver operation is merged with the target stream.

Remote Deliveries with the UCM-ClearQuest Integration

In a project that uses the UCM-ClearQuest integration, the project manager may set the Transition to Complete After Delivery policy on a project (see *Deliveries and the UCM-ClearQuest Integration* on page 102). This policy transitions activities in an associated ClearQuest user database to a Complete state when the deliver operation completes successfully. For this policy to work in a MultiSite environment, all activities being delivered must be mastered by the same PVOB replica that masters the target stream.

In a MultiSite environment that uses the UCM-ClearQuest integration, the project manager sets the Transfer ClearQuest Mastership Before Delivery policy on a project to have the software change the mastership of all activities being delivered. If the deliver operation is local and all activities being delivered are not mastered locally, the deliver operation fails.

For a remote deliver operation, the Transfer ClearQuest Mastership Before Delivery policy causes the following behavior:

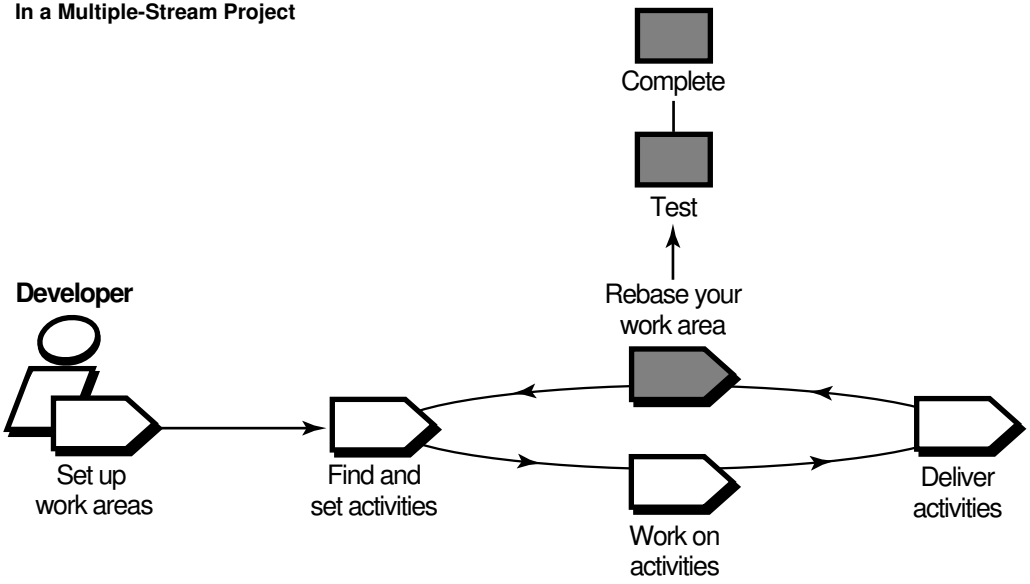
- If all activities are mastered by the remote replica, the deliver operation proceeds.
- If the deliver operation contains activities that are mastered by the local replica, MultiSite transfers mastership of those activities to the remote replica. The project integrator at the remote site completes the deliver operation.

  If the project manager sets the Transfer ClearQuest Mastership After Delivery on the project, MultiSite transfers mastership of the activities back to the local replica.
- If the deliver operations contains activities that are mastered by a third replica, the deliver operation fails.

Rebasing Your Work Area

6

In a Multiple-Stream Project



Periodically, your project manager or project integrator incorporates delivered work into new baselines. Some of these baselines constitute a stable and significant source configuration, and your project manager recommends the baseline.

Rebasing has several benefits:

- You stay up-to-date with other developers' work.
- You reduce the number of merges and minimize the time required to merge versions.
- You identify integration problems early.

Each time your project manager recommends a new baseline for your parent stream, rebase your development work area.

Rebasing involves these tasks:

- Preparing your development view
- Starting the rebase operation

- Testing your development work area
- Completing the rebase operation

Preparing Your Development View

Before you start a rebase operation, check in all files and directories in your development view. You cannot start a rebase operation from a view that contains checkouts.

To Prepare Your Development View

1. In the Rational ClearCase Explorer Folder pane, access the project development view. Right-click the view root directory and then click **Find Checkouts**.
2. In the Find Criteria dialog box, select these criteria:
   - **Include subfolders**
   - **Search only for your checkouts**
   - **Search only for checkouts in the current view**
3. Click **OK**.

   Find Checkouts produces a list of file or directory versions checked out to your development view and indicates which activity each checked-out version belongs to.
4. To see which changes are in a checked-out version:
   - To see the differences between the previously checked-in version and the current version, click **Tools > Compare with Previous Version**.
   - To see checkin comments, click **Tools > History**.
5. Do **one** of the following:
   - To write to the VOB the changes for a given file, check it in. Select the file and click **Tools > Check In**.
   - To undo your changes for a file, cancel its checkout. Select the file and click **Tools > Undo Checkout**.

     During the cancel checkout operation, you can choose to save your changes in a *view-private file*.

Starting the Rebase Operation

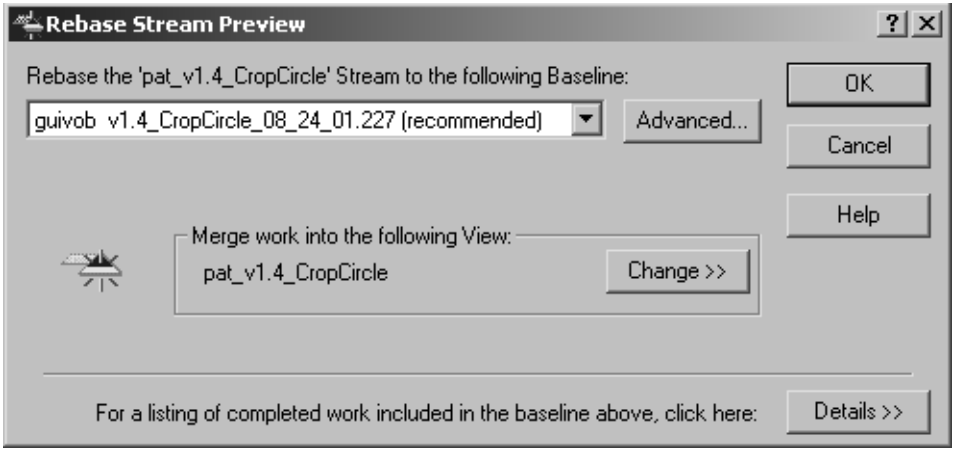
After you prepare your development view (see *Preparing Your Development View* on page 120), you can start the rebase operation. By default, in a multiple-stream project, you rebase your development stream either to baselines recommended by the parent stream or to baselines created by the parent. Typically, these recommended baselines are created by other streams in the project. In a single-stream project, an integrator rebases the integration stream to identify a major project milestone (see *Rebasing in a Single-Stream Project* on page 125).

To Start the Rebase Operation

To start the rebase operation, right-click the root directory of the development view in ClearCase Explorer and click **Rebase Stream**.

The Rebase Stream Preview dialog box opens (Figure 33), showing the name of the stream that you are using.

Figure 33 Rebase Stream Preview Dialog Box



By default, the Rebase Stream Preview dialog box presents the parent stream recommended baseline.

You can also do the following in the Rebase Stream Preview dialog box:

- Verify that the dialog box displays your development view under **Merge work into the following view**. If you attached multiple views to your development stream, click **Change** and choose one view to use for the rebase operation.

You cannot deliver or rebase from other views attached to the development stream until you complete the operation. However, you can check out, check in, and run most other ClearCase commands.

- To see the changes that will be available in your development stream after rebasing to the displayed baseline, click **Details**. The dialog box expands to show the list of activities in the baseline. To see more information about an activity, select it and click **Properties**.

  In the expanded area, select **Show integration activities** to see the baseline integration activities. An integration activity records a change set for a deliver or rebase operation.

- Control merge behavior (see *Merging Versions* on page 124)

  ClearCase tries to do the merge automatically. However, you can perform the merge on each element manually. To handle such a need, select **Merge elements graphically** to turn off all automated merging. If you select this option, the merge operation prompts you for confirmation before proceeding with each change.

To continue the rebase operation, click **OK**. To rebase to baselines other than recommended, see *Rebasing to Baselines Other Than Recommended*.

Rebasing to Baselines Other Than Recommended

The rules for rebasing to baselines other than recommended are different for integration streams and development streams. For an integration stream, you can only rebase to baselines from streams in other projects. The rules differ slightly for modifiable and read-only (nonmodifiable) components.

For a development stream, you can rebase to a baseline that is other than recommended and that is for a modifiable component if the baseline is compatible with your parent stream. You can rebase to a baseline from:

- The parent stream
- The parent stream foundation
- Other streams in the same project if the baselines were delivered to (and are contained in) your parent stream and contain the foundation baseline of your current stream.

The rebase operations ensures that you deliver your changes only from your stream.

If a baseline is for a nonmodifiable component, you are not restricted to selecting a baseline that is compatible with your parent stream. However, a restriction is imposed that you cannot modify elements in the component of that baseline. When you deliver

from your stream, this restriction ensures that you deliver only changes made in your stream and that you do not strand changes in your stream foundation baseline.

The restriction allows you to configure your stream to a baseline of a nonmodifiable component that is not compatible with your parent stream or to a baseline of a component for which the parent stream does not configure a baseline. If you rebase your stream to an incompatible baseline and the component becomes modifiable, you still cannot modify the component until the baseline is made compatible with your parent stream (see *Delivering Changes to a Read-Only Component* on page 113).

To Rebase to Baselines Other Than Recommended

To rebase to baselines other than recommended, start the rebase operation (see *To Start the Rebase Operation* on page 121).

1. In the Rebase Stream Preview dialog box, click **Advanced**. In the Advanced Baseline Configuration dialog box, specify the list of baselines.
2. Click **Add**; in the Add Baselines dialog box, select baselines to be added to the list.
3. Select a listed baseline and do **one** of the following:
   - To remove it from the list, click **Remove**.
   - To modify the baseline, click **Change**. And, in the Change Baseline dialog box, select another baseline.

When you complete the list of baselines, click **OK** to quit the Advanced Baseline Configuration dialog box and click **OK** to continue the rebase operation.

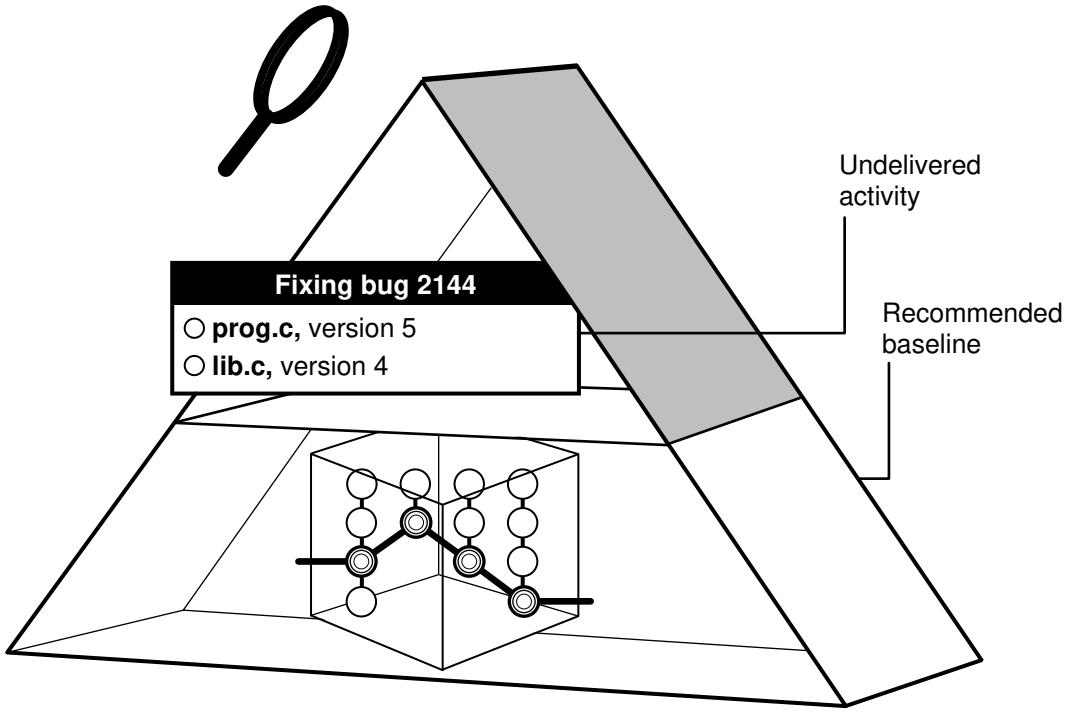
Rebasing Your Development Work Area

In rebasing your work area to the recommended baseline, Rational ClearCase takes the following actions:

- It changes your development stream configuration to select the new recommended baseline. The stream configuration continues to select any activities created in the development stream (Figure 34).
- For development views that are *snapshot views*, it starts an update operation to copy versions in the new baseline into your view.

  If other snapshot views are attached to your development stream, you must update them so that they contain the changes in the stream. Development views that are dynamic views show versions in the new baseline without requiring an update operation.

Figure 34 Development Work Area After Rebasing



As it rebases your development work area, ClearCase displays its progress in the Rebasing Stream dialog box.

To Stop and Resume the Rebase Operation

You can stop the rebase operation temporarily by clicking **Stop** in the Rebasing Stream dialog box. To resume the operation, click **Resume** in the Rebasing Stream dialog box or right-click the root directory of the view in ClearCase Explorer and then click **Rebase Stream**.

Merging Versions

For any version you modify in your development stream, if another team member modified and delivered a version of the same element, you must merge when you rebase to a baseline that contains the delivered version. The following circumstances involve merging:

- As part of working on an activity, you check in a version of prog.c from your development view.

- Another developer checks in and delivers a new version of prog.c to the parent stream.
- Your project manager adds the new version to the recommended baseline.
- You rebase your development stream. As part of the rebase operation, you must merge the other developer's version into your development stream.

As it does during the deliver operation, 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 start Diff Merge so that you can resolve the conflicts. For information about using Diff Merge to resolve conflicting differences, see *Using Diff Merge to Resolve Differences* on page 110.

Rebasing in a Single-Stream Project

Special considerations apply to rebasing in a view belonging to the integration stream of a single-stream project.

If you start the rebase operation from the graphical user interface and rebase detects a checked-out file, a pop-up appears warning you of the conflict. To continue the rebase operation, click **Continue** to resume the rebase or click **Close** to stop the rebase and resume later.

The element logs display additional information about the conflicting checkouts, for example, the view and the name of the user who checked out the file.

If you perform the rebase operation from the command line and rebase detects any checked-out directory or file, the rebase warns you of the conflict. Rebase continues processing the remaining files and merges all directories and files possible. Then the rebase interrupts to allow you to resolve any checkout and merge issues, after which you need to resume the rebase (see *To Stop and Resume the Rebase Operation* on page 124).

Testing Your Development Work Area

After ClearCase changes your stream configuration 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. If your organization uses **clearmake** or **omake**, you can use these ClearCase build tools for your builds; however, the *build auditing* and *build avoidance* features are available only from dynamic views.

For more information, see *Building Software* and the **clearmake** or **omake** reference pages in the *Command Reference*.

In addition, you may need to do the following:

- Check out elements
- Undo a rebase operation

Checking Out Elements

In the process of building and testing, you may need to check out elements in your development view and modify them. As described in *Under the Hood: Integration Activities and Baselines* on page 110, ClearCase creates an integration activity to record any modifications you make during the rebase operation.

> **Note**: We recommend that you postpone checking in your modifications until you complete the rebase operation. Checking in complicates efforts to undo the rebase operation, if you find it necessary to do so.

Undoing a Rebase Operation

At any time before completing the rebase operation, you can undo changes made as a result of the operation. This undo procedure is valid only if you have started a rebase operation and have not completed it.

To Undo a Rebase Operation

1. If you started the Diff Merge tool, close it.
2. Close the Rebase Status dialog box, if open.
3. In ClearCase Explorer, right-click the view root directory and click **Rebase Stream**.
4. In the Rebase Stream dialog box, select **Undo the rebase** and click **OK**.

ClearCase cancels all checkouts in the development view and removes all *merge arrows* created during the current operation. (ClearCase uses merge arrows in VOBs to keep a history of merges.)

If you checked in any versions to the development view, you must remove them by using the **cleartool rmver** command. Because **rmver** erases part of your organizational development history, your organization may not allow individual developers to use it. Be very conservative in using this command. For more information, see the **rmver** reference page in the *Command Reference*.

Completing the Rebase Operation

As with the deliver operation, ClearCase imposes the following state model: rebase operations are either *in progress* or *complete*, and you cannot start a new operation for a development stream when one is in progress.

Completing a rebase operation consists of two tasks: checking in any merge results and changing the state of the operation to Complete.

After testing your work, click **Complete** in the Rebase Status dialog box. ClearCase checks in any versions checked out to your development view and notifies your development stream that the rebase operation is complete. During the rebase completion, the Rebase Status dialog box remains open.

If you closed the Rebase Status dialog box before completing the operation, complete it as follows:

1. In ClearCase Explorer, right-click the view root directory and click **Rebase Stream**.
2. In the dialog box, click **Complete the rebase** and click **OK**.

The rebase operation is placed in the Complete state.

Other Development Tasks

7

Chapter 4, *Working on Activities*, describes tasks that you perform daily or weekly. You may need to perform some of the tasks described in this chapter less often:

- Adding files and directories to source control
- Moving, removing, and renaming elements
- Accessing elements not loaded into a snapshot view
- Adjusting the scope of a view
- Assigning a snapshot view to a drive
- Moving work from a development stream to another project
- Registering a snapshot view
- Moving views
- Accessing views and VOBs across platform types
- Regenerating the view.dat file

Adding Files and Directories to Source Control

You can add files or directories to source control (version control) at any time during the development cycle (see *Components and Baselines* on page 16).

Usually, you add a few files to a directory that is already under source control. Less frequently, you may need to add an entire directory tree to source control.

This section explains the following tasks:

- Adding files and directories to an existing directory
- Adding a directory tree to source control

To Add Files and Directories to an Existing Directory Tree

You can add files and directories to source control to an existing directory tree. Do the following:

1. In Rational ClearCase Explorer, access the view used for your development task (see *Accessing Your Development View* on page 54).
2. Navigate to the parent directory to which you want to add the files or directories.

3 If the files or directories are not present, drag them to the parent directory.

4 Select the files and directories you want to add to source control.

If you specify a comment, upon checkin, the software adds the comment to all of the elements you selected.

If specify a comment and do not select the parent directory in the list of elements to add to source control, the software uses the default ClearCase comment for the parent directory. If you specify a comment and do select the parent directory, the software uses that comment for the parent directory.

5 Right-click one of the selected objects and click **Add to Source Control**.

We recommend you select items that are the farthest from the root of the directory tree: the **Add to Source Control** command for any given file or directory also adds any parent directories (up to the VOB root directory) that are not already *elements*.

The Check Out dialog box appears, with the currently set activity selected in the **Activity** list.

6 To set a different activity (or, if you have not set the view to an activity), do the following:

a Select one from the **Activity** list. To filter the list, click **Browse**.

b If an appropriate activity does not exist, create and set a new one by clicking **New**. If your project is enabled for ClearQuest, choose a record-type from the shortcut menu that appears when you click **New**. ClearCase changes the activity to an Active state type, and ClearCase adds the version you check out to the activity change set.

7 Click **OK**.

Under the Hood: What Happens When You Add to Source Control

The **mkelem** or **Add to Source Control** command always creates an element and initializes its version tree by creating a single branch (named **main**) and a single, empty version (version 0) on that branch. The following arguments for the **mkelem** command or options in the Add to Source Control dialog box determine optional ClearCase behavior:

- Selecting **Keep checked out** or using **mkelem** with no arguments checks out the element. Any view-private data that corresponds to the element path remains in your view only and is added to version 1 in the VOB when you check in (Figure 35).
- Clearing **Keep checked out** or using **mkelem –ci** checks in the element, using any existing view-private data that corresponds to the element path as the content for

version 1. Your view config spec determines the branch on which ClearCase creates version 1.

- Using **mkelem –nco** suppresses automatic checkout; **mkelem** creates the new element, along with the **main** branch and version **\main\0**, but does not check it out. If *element-path* exists, it is moved to a .keep file.
- (Replicated VOBs only) Selecting **Make current replica the master of all newly created branches** (file elements only) or using **mkelem –master** or **mkdir –master** assigns to your *current replica* mastership of all branches created during element creation. You will be able to create new versions on the branches.

  Clearing this check box or using **mkelem** or **mkdir** without the **–master** option assigns mastership of a new branch to the VOB replica that masters the associated branch type. If this replica is not your current replica, you cannot create new versions on the branch.

  You can set the default for mastership assignment in the User Options dialog box:

  a In ClearCase Explorer, click **Tools > Options**.

  b In the User Options dialog box, click **ClearCase Options**.

  c In the ClearCase User Options dialog box, click the **Operations** tab.

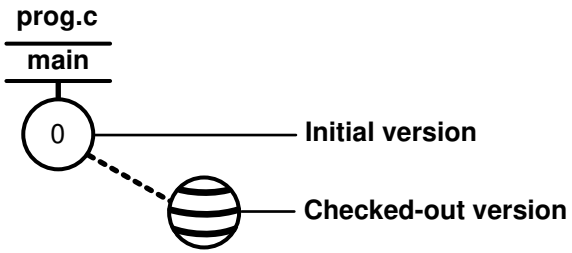
  d Click **Advanced Options**.

  e Select **When creating an element in a replicated VOB, make current replica the master of all newly created branches**.

  f Click **OK**.

Other views do not see the element until you check in the parent directories of the element (the **Add to Source Control** command does this for you) and check in the file or directory.

Figure 35 Creating an Element



Note: *VOB links* make it possible to have more than one copy of a directory in a snapshot view. When you add a file or directory to a snapshot view, ClearCase updates all other instances of the parent directory that are loaded into your view.

To Add Elements for a New Development Task

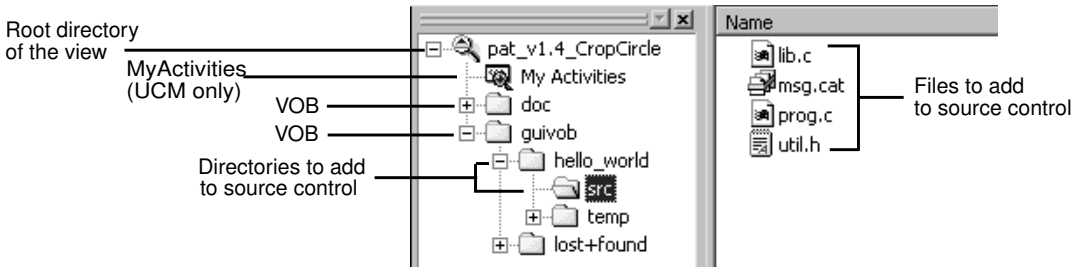
Note: This procedure assumes that a VOB exists. For information about creating VOBs, see *Managing Software Projects*.

1. In ClearCase Explorer, access the view you created for the development task (see *Accessing Your Development View* on page 54).
2. If you work in a snapshot view, create a folder in the view that corresponds to each VOB to which you want to add files and directories.

   If you work in a dynamic view, make sure that each VOB to which you want to add files and directories is accessible from your view. If a VOB is not accessible, activate it on your computer (see *To Activate VOBs* on page 138).
3. Under each VOB, do **one** of the following:
   - Create a directory structure.
   - If files and directories already exist, copy them into the appropriate location within the directory tree.

   When you are finished, the view resembles Figure 36.

Figure 36 Directory Structure in a Snapshot View



4. In the Details pane of ClearCase Explorer, select files and directories.

   We recommend that you select items that are the farthest from the root of the directory tree: the **Add to Source Control** command for any given file or directory also adds any parent directories (up to the VOB root directory) that are not already *elements*. For example, in Figure 36, if neither \guivob\hello_world nor \guivob\hello_world\src are elements, you can add both directories to source control by adding \guivob\hello_world\src to source control.

5 Right-click one of the selected objects. On the shortcut menu, click **Add to Source Control**.

Importing Files

If you add a large number of files and directories to source control, use the **clearfsimport** command (or **clearexport** command) and **clearimport** command. For more information, see the **clearfsimport** and **clearimport** reference pages in the *Command Reference*.

Moving, Removing, and Renaming Elements

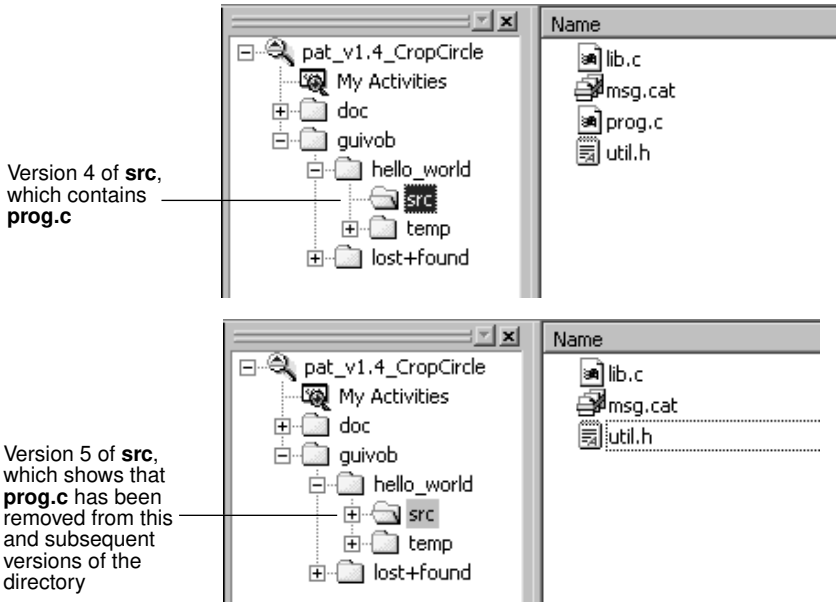
This section explains how to move, remove, and rename elements.

Moving and Removing Elements

Because directories as well as files are under ClearCase control, you can move or remove elements from specific versions of directories without affecting the element itself. Moving or removing elements creates new versions of the parent directories to record the modifications.

For example, version 4 of \gui_vob\hello_world\src contains an element named prog.c. If you remove prog.c from the hello_world\src directory, ClearCase creates version 5 of \gui_vob\hello_world\src, which does not contain the prog.c file element. The element prog.c itself is not modified (Figure 37).

Figure 37 Removing an Element Name from a Directory



Before you move or remove an element name from a directory, verify with your project manager that your changes will not adversely affect other team members or break projectwide builds.

To Move an Element Within a VOB

Do **one** of the following:

- In the ClearCase Explorer Details pane, drag an element to its destination directory in the Folder pane.
- In the ClearCase Explorer Details pane, right-click an element and click **Cut**. Then, right-click the destination directory and click **Paste**.

Moving an Element to Another VOB

You cannot move an element in a UCM component VOB to another VOB.

To Remove an Element Name from a Directory

In the ClearCase Explorer Details pane, right-click an element and click **Delete**.

Other Methods for Removing Elements

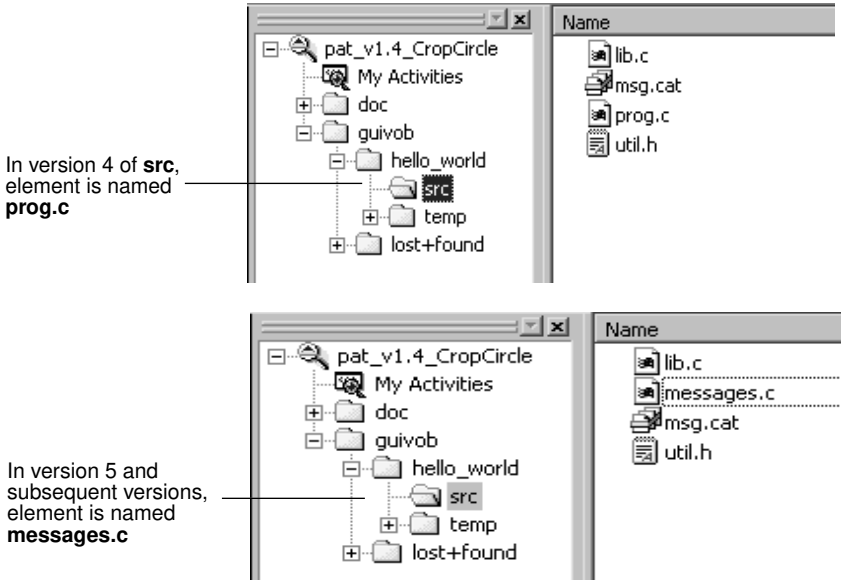
Removing an element from its parent directory does not affect the element itself. But two other types of a removal operation do irrevocably affect an element, and we recommend that you be very conservative in using these operations:

- Removing a version from a version tree. For more information, see the **rmver** reference page in the *Command Reference*.
- Removing an element from a VOB. For more information, see the **rmelem** reference page in the *Command Reference*.

Renaming Elements

Renaming an element creates a new version of the parent directory to catalog the new element name. The element uses its new name in subsequent versions of its parent directory, but previous versions of the parent directory refer to the element by its previous name (Figure 38).

Figure 38 Renaming an Element



Before you move or remove an element name from a directory, verify with your project manager that your changes will not adversely affect other team members or break project builds.

To Rename an Element

1. In the ClearCase Explorer Details pane, right-click an element and click **Rename**.
2. In the Details pane, type a new name for the element
3. Press ENTER.

Accessing Elements Not Loaded into a Snapshot View

While working with source files in a snapshot view, you may need to see the contents of elements that are not loaded into the view or see ClearCase information about these nonloaded elements. For example, you may have chosen not to load a VOB that contains functional-specification documents. However, you may want to check periodically whether the functional specifications have been modified by reviewing the element history.

Listing All Elements in the VOB Namespace

You can set up ClearCase Explorer to provide a directory listing of nonloaded file and directory elements. ClearCase Explorer displays the version of the element that would be selected if the element were loaded in the view. The Version column in the Details pane indicates which version the view is selecting. Then, you can display the history, version tree, and properties sheet of the nonloaded element, and compare versions of the element.

To See Nonloaded Elements from ClearCase Explorer

1. In ClearCase Explorer, click **Tools > Options**.
2. In the Options dialog box, under **Snapshot View Options**, select **Show Unloaded Snapshot View Elements**.
3. Click **OK**.

   The Details pane uses an icon and the State column to indicate which versions are not loaded.

To See Metadata for Nonloaded Versions

Metadata is information that ClearCase keeps about elements under source control. To see metadata for a nonloaded version, click the version in the Details pane and select commands from the **Tools** menu.

Viewing the Contents of a Nonloaded Version

You can view nonloaded files or copy them into your view for build purposes, but you cannot check them out. Only file elements that are loaded into the view can be checked out.

To see the contents of a nonloaded version, you can do either of the following:

- Use the **Send To** command from the shortcut menu of the version in the Version Tree Browser. The **Send To** command uses the set of shortcuts in your Send To folder, which is a function that the Windows operating system provides. For information about your Send To folder, see Windows help.
- Copy the version into your view with the **cleartool get** command.

  Note: You cannot use the **Send To** command or **cleartool get** for directory elements.

To Copy a Nonloaded Version with cleartool get

1. In the ClearCase Explorer Details pane, in the Version column, note the *version-extended path* of the unloaded element, which is displayed in the Version column.
2. Right-click the parent directory and from the shortcut menu, click **Command Prompt** to display a **cleartool** prompt.
3. Type the **cleartool get** command and use the following syntax:

   get –to *filename version-extended-path*

   For example, **get –to** prog.c.previous.version prog.c@@\main\v3.1_fix\10 copies prog.c@@\main\v3.1_fix\10 into your view under the name of prog.c.previous.version.

Adjusting the Scope of a View

At any time during a development cycle, you may need to change the set of elements available to your view. For example, if you work in a snapshot view, you can reduce the amount of time required for deliver and rebase operations by narrowing the scope of your view to include only the set of elements related to your activities. (The fewer elements in your snapshot view, the less time needed to update it.) Note, however, that to test your work during the deliver operation, your view must include any files required to satisfy build dependencies.

To adjust the scope of a view, you can do the following:

- Change which elements are loaded into a snapshot view
- Activate or deactivate VOBs for dynamic views

Adjusting the scope of a view does not affect the stream to which the view is attached or any other views that are attached to the stream. For example, if a recommended baseline includes a new set of files and directories, rebasing your development stream makes the new files and directories available to your stream, but they may not be visible in your development view, depending on the view scope.

To Change Which Elements Are Loaded into a Snapshot View

1. In the ClearCase Explorer Shortcut pane, click **Toolbox**. Then click **Base ClearCase > Edit View Properties**.
2. In the Edit View Properties dialog box, select the view and click **Edit**.
3. In the Properties dialog box, click the **Load Rules** tab.
4. On the **Load Rules** tab, click **Edit Load Rules**.
5. In the Choose Elements to Load dialog box, click **Add** to load another element into the view, or click **Remove** to unload a file or directory from the view.
6. Click **OK**.

Activating or Deactivating VOBs for Dynamic Views

Activating a VOB on your computer makes its files and directories available to your dynamic views. Deactivating a VOB frees your computer mount points.

To Activate VOBs

Note: Usually, ClearCase mounts all VOBs that contain versions of elements in a project baseline when you start a dynamic view that is attached to a project stream. However, you may need to follow this procedure if you deactivate a VOB and then want to remount it.

1. On the Windows desktop, click **Start > Programs > Rational Software > Rational ClearCase > Administration > Mount VOB**.
2. In the Mount VOB dialog box, select the VOBs containing your source files.
3. Select **Reconnect at Logon** to activate the VOBs when you log on.
4. Click **OK**.

To Deactivate VOBs

1. On the Windows desktop, click **Start > Programs > Rational Software > Rational ClearCase > Administration > Unmount VOB**.

2 In the Unmount dialog box, select one or more VOBs.

3 Click **Unmount**.

Assigning a Snapshot View to a Drive

If your makefiles or other files require absolute paths with a specific drive, map the folder for your snapshot view to a drive.

Depending on your computer configuration, two methods are available:

- Make the root directory of the snapshot view shared and then assign it to a drive.
- Use the **subst** command.

Assign Only the Root Directory. You must assign only the snapshot view root directory to a drive. The root directory of the snapshot view contains a hidden file named view.dat. ClearCase searches for this file to determine whether the current directory is in a snapshot view.

If you assign a directory below the view root to a drive, ClearCase cannot find the view view.dat file and assumes that the current directory is not a snapshot view.

Making a Directory Shareable and Assigning a Drive

From ClearCase hosts running either Windows NT or Windows 2000, you can make a snapshot view root directory shareable (see *To Enable Sharing on the Snapshot View Root Directory*) and assign it to a drive (see *To Assign the Shared Snapshot View Root Directory to a Drive*). With this method, other team members can access your snapshot view through either My Network Places or Network Neighborhood, but the performance is slightly slower than if you use the **subst** command to assign the view to a drive (see *Using the subst Command for Snapshot View Access* on page 140).

From ClearCase hosts running Windows 98, you can assign a shared directory to a drive only if the directory is on some other host; for example, you can use this method to assign someone else's snapshot view to a drive.

To Enable Sharing on the Snapshot View Root Directory

Note: If you want to share a directory on a Windows 98 computer (for example, so team members can assign your view to a drive on their computers), you must use this procedure to enable file sharing.

1 In Windows Explorer, click the snapshot view root directory; then click **File > Sharing**.

2 On the **Sharing** tab, click **Shared As**.

3 For the sake of consistency, accept the default value in the Share Name box. (Make the share name the same as the leaf name of the snapshot view root directory.)

4 Click **OK**.

Other team members can now assign a drive on their computers to this snapshot view root directory (see *To Assign the Shared Snapshot View Root Directory to a Drive*) or can create a shortcut to the view in ClearCase Explorer (see *To Access Another's Snapshot View from ClearCase Explorer* on page 142).

To Assign the Shared Snapshot View Root Directory to a Drive

If the root directory of the snapshot view is shareable (see *To Enable Sharing on the Snapshot View Root Directory*), you can assign the directory to a drive on your computer.

Note: If you prefer to use the command line, you can use the **net use** command.

1 From Windows Explorer, click **Tools > Map Network Drive**.

2 In the Map Network Drive dialog box, in **Drive**, select a drive.

3 In **Folder**, enter the path to the shared folder of the snapshot view root directory or click **Browse** and navigate to the view root directory in My Network Places or Network Neighborhood.

4 Click **Finish**.

Using the subst Command for Snapshot View Access

Assigning a snapshot view root directory to a drive with the **subst** command provides slightly better performance than making the snapshot view a shared directory (see *Making a Directory Shareable and Assigning a Drive* on page 139). However, only shared directories are accessible through My Network Places or Network Neighborhood. (To assign your own view to a drive on Windows 98 computers, you must use the **subst** command.)

1 Open a command shell.

2 Enter **subst** *driveletter***:** *view-path*. For example, the command

```
subst Y: C:\library\pat_v1.4_cropcircle
```

maps the pat_v1.4_cropcircle snapshot view root directory to drive Y.

Assign only the view root directory to a drive. For more information about the **subst** command, type **help subst** in a command shell.

Moving Work from a Development Stream

Under normal circumstances, project managers and integrators use baselines to share work between projects. But if you start work on an activity in your development stream and then determine that you cannot complete the work for the current project, we recommend that you move the work to the project in which you intend to complete it. To facilitate the merge process, move the work to the new project as soon as you determine that you cannot complete it in the original project. (The longer you wait to merge the work, the less you may remember about it and the greater the potential for merge conflicts.)

We recommend against working in a development stream that contains activities you do not intend to complete for the current project; doing so may prevent the rebase operation from synchronizing your development stream with the work in the baseline.

To Move Work from a Development Stream to Another Project

1. Get the activity-selector of the activity:
   a. Find the activity. For example, use **My Activities** in ClearCase Explorer (see *Finding Activities* on page 62).
   b. Right-click the activity and, on the shortcut menu, click **Properties**.

      On the **General** tab of the Properties dialog box, in **ID**, the activity-selector of the activity is displayed.
   c. Write down the activity-selector or copy it to the clipboard.
   d. Quit the Properties Browser.
2. Join the project in which you intend to complete the work. (See Chapter 2, *Setting Up Work Areas*.)
3. Decide whether to add the work to the new project integration stream or to another development stream in the new project (see *Integration Stream* on page 45).
4. From your view in the original project, deliver the work to a nondefault target (the stream in the new project in which you intend to complete the work). (See *Delivering to a Nondefault Target* on page 103.)

   To specify the work to be delivered, use the activity selector that you obtained in Step 1.
5. Lock, in the obsolete state, your development stream for the original project.
6. Remove your development view for the original project.

Registering a Snapshot View

If you need to perform ClearCase operations in a snapshot view that you did not create or have not updated, you must register that view.

When you create or update a snapshot view or click a view shortcut in ClearCase Explorer, ClearCase registers the snapshot view in your computer registry. The ClearCase integration with ClearCase Explorer and Windows Explorer recognizes as ClearCase objects the files and directories below a registered snapshot view. If the snapshot view that ClearCase is trying to access is not registered on your computer, you must explicitly register the view.

If a snapshot view is not registered by you and you access files and directories under the view root directory through Network Neighborhood, the ClearCase integration with Windows Explorer does not recognize the files or directories as ClearCase objects. For example, if you use Windows Explorer to navigate to a team member's snapshot view on a different computer and right-click a file in the view, ClearCase options are not available. If you try to use an unregistered snapshot view to perform ClearCase operations (for example, as the target of a deliver operation or as the source for a rebase operation), ClearCase cannot find the view.

To perform ClearCase operations on the files and directories in a snapshot view that is not registered by you (for example, a team member's snapshot view), you must register the snapshot view. You can use ClearCase Explorer (see *To Access Another's Snapshot View from ClearCase Explorer* on page 142) or Windows Explorer (see *To Register a Snapshot View Using Windows Explorer* on page 143) to register the snapshot view.

To Access Another's Snapshot View from ClearCase Explorer

To access a snapshot view that is not registered by you, create a view shortcut in ClearCase Explorer. If the root directory of the snapshot view is shareable (see *To Enable Sharing on the Snapshot View Root Directory* on page 139), follow these steps on your computer:

1. On the **Views** tab of the Shortcut pane, right-click and click **Add View Shortcut**.
2. In the Add a New View Shortcut dialog box, in **View Type**, select Snapshot.
3. In **Page**, select the location in which the place the shortcut icon.
4. In **Description**, enter text for the ToolTip.
5. In **Snapshot Location**, do either of the following:
   - Enter the path to the shareable snapshot view root directory.

- Click the browse button to the right of the **Snapshot Location** text box; navigate to and select the shareable root directory under My Network Places or Network Neighborhood; and click **OK**.

6 Click **OK**.

7 Click the view shortcut.

This registers the view for your use on any Windows computer if you use a roaming user profile. If you use a local user profile, this registers the view for your use on this one computer. Click the view shortcut to access the snapshot view.

Note: You cannot register a snapshot view that was created from a UNIX host (see *Accessing Views Across Platforms of Different Types* on page 146).

To Register a Snapshot View Using Windows Explorer

To use Windows Explorer to perform ClearCase operations on the files in a snapshot view that is not registered by you, register the view. If the root directory of the snapshot view is shareable (see *To Enable Sharing on the Snapshot View Root Directory* on page 139), proceed as follows:

1 In Windows Explorer, navigate to the root directory of the snapshot view that you want to register.

2 Right-click the root directory of the view.

This registers the view for your use on any Windows computer if you use a roaming user profile. If you use a local user profile, this registers the view for your use on this one computer.

Note: You cannot register a snapshot view that was created from a UNIX host (see *Accessing Views Across Platforms of Different Types* on page 146).

Moving Views

This section discusses the following tasks:

- Changing the physical location of a snapshot view directory tree
- Moving a view storage directory

For information about changing a view tag, see the **mktag** reference page in the *Command Reference*.

Changing the Physical Location of a Snapshot View

If the snapshot view storage directory is in a storage location, you can use standard Windows commands (such as **Cut** and **Paste** commands or drag and drop operations in Windows Explorer) to move it. You can move the view to a different computer, but the computer must run a Windows operating system.

Caution: If the view storage directory is located below the root directory of the view, **do not use** standard Windows commands to move the snapshot view. Instead, see *Moving a View Storage Directory*.

To Find the Location of the View Storage Directory

1. In ClearCase Explorer, right-click the view root directory and click **Properties of View**.
2. In the View Properties dialog box, click **Advanced**.

   The **Host Path** box displays the path for the view storage directory.

Update After Moving

After moving a snapshot view, you must update any view shortcuts in ClearCase Explorer that use the old path to the view. If you did not create a ClearCase Explorer shortcut for the view and you move a view to a different computer, you must register its new location. Until you register its new location, the ClearCase user interface continues to present the old location.

Moving a View Storage Directory

Each dynamic view and snapshot view includes a view storage directory, which ClearCase uses to maintain the view. **Do not use** standard Windows commands (such as **Cut** and **Paste** commands or drag and drop operations) to move a view storage directory for the following reasons:

- The view storage directory includes a database. Moving the database without first shutting down the *view_server* process of the view can corrupt the database.
- ClearCase stores the location of view storage directories in its own set of registries. The information in these registries must be correct for you to perform ClearCase operations in your views. In a dynamic view, the location in ClearCase registries must be correct for you to access any file or directory in the view.

We suggest that you ask your ClearCase administrator to move view storage directories because it may affect other, potentially many other, ClearCase users at your

site. The *Administrator's Guide* for Rational ClearCase describes the procedure for moving view storage directories.

Caution: You will lose data (including view-private files in a dynamic view) if you move a view storage directory without following the procedure described in the *Administrator's Guide*.

Accessing Views and VOBs Across Platform Types

ClearCase supports environments in which some ClearCase hosts use a Microsoft Windows operating system and others use a UNIX operating system.

This section discusses the following topics:

- Creating views across platform types
- Accessing views and VOBs across platform types
- Developing software across platform types

Creating Views Across Platform Types

Your ClearCase administrator can set up storage locations on Windows and UNIX server hosts. Any snapshot view that you create can use one of these storage locations, regardless of the platform type of the server host. For more information about storage locations, see the **mkstgloc** reference page in the *Command Reference*.

For a dynamic view, the view storage directory must be located on a host of the same platform type as the host from which you create the view. If you create a dynamic view from a UNIX host, you must locate the view storage directory on a ClearCase host on UNIX; if you create a dynamic view from a Windows host, you must locate the view storage directory on a ClearCase host that runs Windows software and is set up to store view storage directories. We recommend that you locate dynamic view storage directories on the host from which you most often use the view.

Snapshot View Characteristics and the Operating System

For snapshot views, the operating system from which you create the view determines view characteristics; the operating system that hosts the files and processes related to a snapshot view do not affect the view behavior.

For example, it is possible to create a snapshot view from a Windows host and locate the view directory tree and the view storage directory on a ClearCase host on UNIX (assuming that you use third-party software to access UNIX file systems from Windows computers). Although all files related to the view are on a UNIX workstation,

because you created the view from a Windows host, the view behaves as if its files are located on a Windows computer:

- Case-sensitive file lookups are not performed in the view.
- The view does not create symbolic links if the load rules encounter a VOB symbolic link.
- You can issue ClearCase commands for the view only from Windows hosts. (ClearCase hosts on UNIX do not recognize the directory tree as a snapshot view.)

Accessing Views Across Platforms of Different Types

This section describes support for accessing a view residing on a platform that differs from the platform from which it is being accessed.

Accessing UNIX Snapshot Views from Windows Hosts

ClearCase supports a set of third-party products for accessing UNIX file systems from Windows computers. If your organization uses one of these products, you can access UNIX snapshot views from Windows Explorer (or a command prompt) just as you would access any other directory tree on a UNIX workstation.

You can access snapshot views across platforms, but you cannot issue ClearCase commands across platforms. For example, you cannot check out files in UNIX snapshot views from Windows hosts nor can you create shortcuts to UNIX snapshot views from ClearCase Explorer.

If, from a Windows host, you hijack a file in a UNIX snapshot view, ClearCase detects the hijack when you update the view from a ClearCase host on UNIX.

Accessing Windows Snapshot Views from UNIX Hosts

ClearCase does not support accessing Windows file systems from UNIX workstations.

Accessing UNIX Dynamic Views from Windows Hosts

ClearCase supports a set of third-party products for accessing UNIX file systems from Windows computers. If your organization uses one of these products, you can complete the following tasks to access UNIX dynamic views from Windows computers:

1 On the UNIX system, create the UNIX view with the proper text mode. For more information, see *Developing Software Across Platforms of Different Types* on page 148.

2 On the Windows system, use Region Synchronizer to import the UNIX view tag of the view into your Windows network region.

3 In ClearCase Explorer, refresh view shortcuts. ClearCase Explorer detects the view tag, starts the view, and creates a view shortcut for you.

Accessing Windows Dynamic Views from UNIX Hosts

ClearCase does not support products for accessing Windows file systems from UNIX workstations. You cannot access Windows views from UNIX hosts.

Accessing VOBs Across Platforms of Different Types

Your ClearCase administrator sets up VOBs on hosts running Windows or UNIX software and creates *VOB tags* in each ClearCase network region that needs to access the VOBs. (For information about registering UNIX VOB tags on a Windows computer, see the *Administrator's Guide* for Rational ClearCase.) Then, from any ClearCase host running Windows or UNIX software, you can create snapshot views to load elements from VOBs that have tags in your network region.

From dynamic views and snapshot views on a ClearCase host on a Windows computer that supports dynamic views, you can access VOBs on Windows computers and UNIX workstations. To access VOBs on UNIX workstations from dynamic views on Windows computers, you must use third-party software that provides access to UNIX file systems from Windows computers.

From dynamic views on a ClearCase host on a UNIX workstation, you cannot access VOBs on Windows computers. Table 3 summarizes your options for accessing VOBs across platforms of different types.

Table 3 ClearCase VOB Access Across Different Platforms

ClearCase host platform	Platform on which VOB is located	Type of view from which you can access source files
Windows computer	Windows computer or UNIX workstation	Snapshot view or dynamic view
UNIX workstation	Windows computer	Snapshot view
UNIX workstation	UNIX workstation	Snapshot view or dynamic view

Developing Software Across Platforms of Different Types

To use a UNIX view on a Windows host to access text files, you need an interop text-mode view. Before a VOB can be accessed from an interop text-mode view, the system administrator must enable the VOB for interop text-mode support.

If developers check in source files from views created on hosts running either Windows and UNIX, consider creating your views in **insert_cr** or **strip_cr** text mode. The text modes change how a view manages line terminator sequences. For more information about choosing a text mode for a view, see the Rational ClearCase Help.

Regenerating the view.dat File

The root directory of a snapshot view contains a hidden file, view.dat. If you delete this file inadvertently, ClearCase no longer identifies the view as a ClearCase object, and you can no longer perform ClearCase operations on files or directories loaded in the view.

To Regenerate the view.dat File

1. Open a command shell.
2. Type this command:

   ccperl *ccase-home-dir***\etc\utils\regen_view_dot_dat.pl**
   ^ [**–tag** *snapshot-view-tag*] *snapshot-view-path*

   For example:

   ccperl c:\rational\clearcase\etc\utils regen_view_dot_dat.pl
   ^ –tag pat_v1.4_cropcircle
   ^ c:\pat_v1.4_cropcircle

   If the view storage directory is under the root directory of the view, you do not need to use the **–tag** *snapshot-view-tag* argument.

Working in a Snapshot View Without the Network

A

If you need to work with your source files from a computer that is disconnected from the network of Rational ClearCase hosts and servers, you can set up a snapshot view for disconnected use.

This chapter describes the following tasks:

- Setting up a view for your hardware configuration
- Preparing the view
- Disconnecting the view
- Working in the view
- Reconnecting to the network
- Using the Update Tool

Note: While disconnected from the network, you cannot access ClearCase information about the files in your view or issue most ClearCase commands. If you want to work from a remote location and continue to access ClearCase information and issue ClearCase commands, consider using the Web interface for ClearCase. Ask your ClearCase administrator whether the Web interface for ClearCase has been configured at your site and what URL you need to supply to your Web browser to access it. For information about using the Web interface, see the Web interface Help.

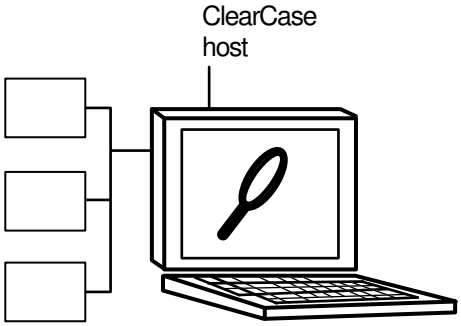
Setting Up a View for Your Hardware Configuration

You can use one of several hardware configurations to work in a snapshot view that is disconnected from the network.

This chapter describes the following recommended configurations:

- Creating and using the view on a laptop computer that periodically connects to the network (Figure 39).

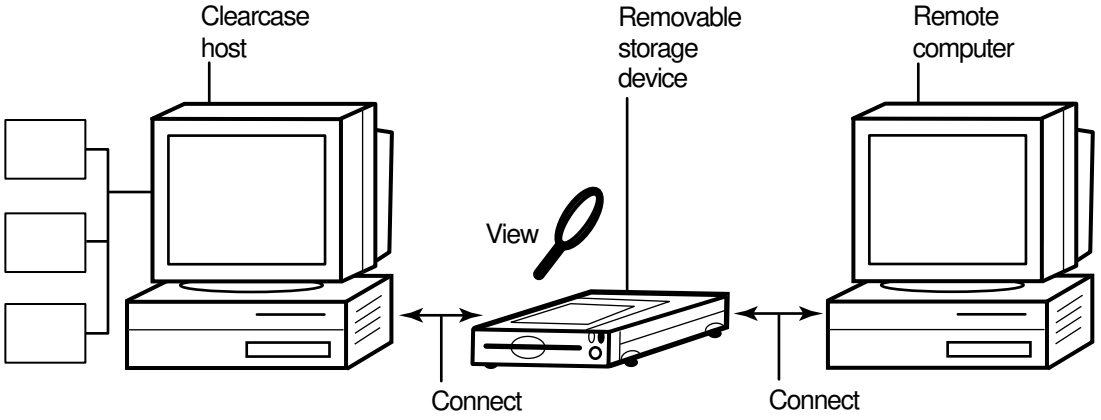
Figure 39 View on a Laptop



Note: The laptop computer must run a Windows operating system.

- Creating and using the view on a removable storage device such as an external hard drive or some other device (such as a Jaz drive) that provides satisfactory read/write performance (Figure 40).

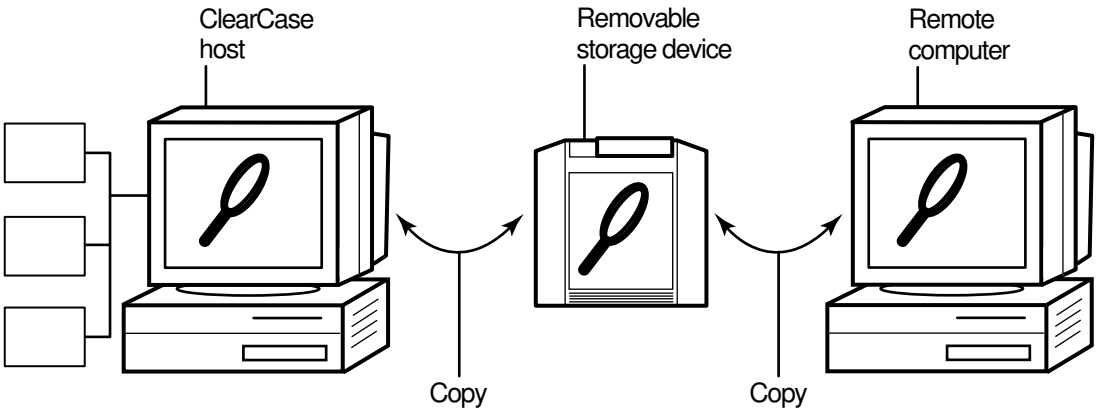
Figure 40 View On a Removable Storage Device



Note: The remote computer must run a Windows operating system.

- Copying the view from a ClearCase host to a temporary, removable storage device such as a diskette or a tape drive, which usually does not provide satisfactory read/write performance, and then copying the view from the storage device to a computer that is disconnected from the network (Figure 41).

Figure 41 Copy the View



Note: The remote computer must run a Windows operating system.

Under the Hood: View Storage in Disconnected-Use Configurations

In all the configurations recommended for disconnected use, the snapshot view storage directory is in a server storage location. We recommend (and in the case of a removable storage device, we enforce) this configuration because the *view_server* process of a view runs on the host that contains the view storage directory or on a host that can access a location on a supported Network Attached Storage (NAS) device which contains the view storage directory.

A view_server is a long-lived process that manages activity for a specific view. If the view storage directory is in the root directory of the snapshot view and you disconnect the view from the network while the view_server process is writing to the storage directory, you can corrupt the data ClearCase uses to maintain your view.

Preparing the View

Before you disconnect the view from the network, complete these tasks:

- Update the view to establish a checkpoint. (For information about updating the view, see *Updating the View* on page 158.)
- Check out the files you expect to modify. After you are disconnected from the network, you cannot check out files, although there are workarounds. (See *Hijacking a File* on page 154.)

When you are no longer connected to the network, you cannot use most ClearCase commands. At this point, the disconnected computer does not distinguish a snapshot view directory from any other directory in the file system.

Disconnecting the View

How you disconnect the view depends on which hardware configuration you use. This section describes disconnecting the view for the following hardware configurations:

- View on a laptop
- View on a removable storage device
- View Copied to a storage device

Hardware Configuration: View on a Laptop

If the view is located on a laptop, you must deactivate the ClearCase-Windows Explorer integration and other developer tools before you disconnect the view. Most ClearCase operations communicate with various ClearCase servers. Even if you install view server software on your host, almost any time you start a ClearCase operation (even an operation as simple as displaying the ClearCase shortcut menu in Windows Explorer), your host must connect to the license server for ClearCase. The ClearCase integration with Microsoft Visual Studio attempts to connect to the license server periodically. If you are disconnected from the network, the application that attempts to connect to the license server for ClearCase locks for several seconds until it returns an error message. Although the application does not fail, such interruptions are unnecessary.

To Deactivate ClearCase Integrations

1. In Control Panel, double-click **ClearCase**.
2. Click the **Options** tab and clear **Connected to network for ClearCase operations**.

3 Disconnect the laptop from the network.

Deactivating the integrations removes ClearCase commands from the ClearCase Explorer and Windows Explorer shortcut menus.

Hardware Configuration: View on a Removable Storage Device

If the view is located on a removable storage device, disconnect the removable storage device from the network computer and reconnect the media to the remote computer.

Hardware Configuration: View Copied to a Storage Device

If you do not have a storage device with satisfactory read/write performance, use the **xcopy** command to copy files from your view to the storage media and from the storage media to the remote computer as follows:

xcopy *path\snapshot-view-directory destination* [**/D** [**:***m-d-y*] | **/M**] **/S /K /R**

For example, **xcopy c:\Library\pat_v1.4_cropcircle e:\temp /D /S /K /R**

The command options accomplish the following:

- The **/D** option instructs **xcopy** to copy only the files that have changed, which is useful if you frequently alternate between the network computer and the nonnetwork computer. Use **/D** if you configured the Update Tool to set the time stamp of updated files to the time at which the files are loaded into the view. You can view or change the Update Tool time stamp option on the **Advanced** tab of the Start Update dialog box. For more information, see Step 3 of *Updating the View* on page 158.

  If you set the Update Tool to create time stamps based on the version-creation time, consider using the **/M** option, which causes **xcopy** to copy the files with the archive attribute set. The **/M** option is less reliable than the **/D** option, because some applications alter the archive attribute without writing the file.
- The **/S** option instructs **xcopy** to copy all subdirectories of the snapshot view.
- The **/K** option keeps the read-only file attribute as established by ClearCase.
- The **/R** option allows **xcopy** to overwrite read-only files.

Working in the View

You cannot use most ClearCase commands when disconnected from the network. Yet you may need to work on files that you did not check out or locate files you have modified. This section provides workarounds for these ClearCase operations.

Hijacking a File

If you need to modify a loaded file element that you have not checked out, you can *hijack* the file. ClearCase considers a file hijacked when you modify it without checking it out. For more information, see *Under the Hood: Determining Whether a File Is Hijacked* on page 158.

When you reconnect to the network, you use the Update Tool to find the files you hijacked. You can do the following with a hijacked file:

- Check out the file. You can then continue to modify it and, when you are ready, check in your changes.
- Undo the hijack. For more information, see *To Undo a Hijack* on page 157.

To Hijack a File

1. In Windows Explorer, right-click the file to hijack and click **Properties** to display the property sheet of the file.
2. On the **General** tab, clear **Read-only**.

To Find Modified Files While Disconnected

You can use the Search applet from either ClearCase Explorer or from **Start > Search > For Files or Folders** to find all files that have been modified after a specified date.

1. Do either of the following:
   - In ClearCase Explorer, right-click on a directory and click **Search**. The directory path is seeded in the **Look in** box.
   - Click **Start > Search > For Files or Folders**.

   The Search Results window appears.
2. In the Search pane, specify the path to the view or to a specific directory in the view.
3. Click **Search Options** and **Date** to define your search criteria for files that have been modified.
4. Click **Search Now**.

   The Details pane expands to display the files and directories it finds.

Reconnecting to the Network

When you return to the office and can access the network directly, do one of the following, depending on your hardware configuration.

Hardware Configuration: View on a Laptop

If you use the view on a laptop, connect the laptop to the network. If ClearCase is installed on the laptop, activate the ClearCase integrations as follows:

1. Click **Start > Settings > Control Panel** and double-click **ClearCase**.
2. Click **Options** and select **Connected to network for ClearCase operations**.
3. Click **OK**.

Hardware Configuration: View on a Removable Storage Device

If you use the view on a removable storage device, connect the removable storage device to the computer on the network.

Hardware Configuration: View Copied to a Storage Device

If you copied the view onto removable media, use **xcopy** with the **[/D | /M] /S /K /R** options to copy files back to the original location on the network computer.

Using the Update Tool

When you are connected to the network, use the Update Tool for the following tasks:

- Determine how to handle hijacked files
- Update the view

Determining How to Handle Hijacked Files

Handling hijacked files involves the following tasks:

- Finding hijacked files
- Comparing a hijacked file to the version in the VOB
- Checking out a hijacked file
- Merging changes to a hijacked file
- Undoing a hijack
- Choosing other ways to handle hijacked files

To Find Hijacked Files

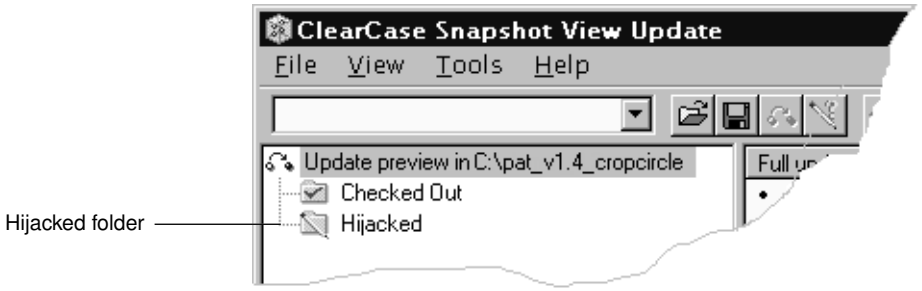
1 In ClearCase Explorer, right-click the root directory of the snapshot view.

2 On the shortcut menu, click **Find Modified Files**.

Select **Display results when closed** and quit the Modified File Search Complete window.

3 If any hijacked files are in your view, the Snapshot View Update window displays a folder in the left pane titled **Hijacked** (Figure 42). Select **No** for the option asking whether you want to check out the hijacked files now.

Figure 42 Hijacked Files in the Results Window



To Compare a Hijacked File to the Version in the VOB

You can use the Diff Merge tool to see how the hijacked file differs from the checked-in version of the file:

1 In the right pane of the Snapshot View Update window, right-click a hijacked file.

2 On the shortcut menu, click **Compare with Original Version**. For information about using the Diff Merge tool, see the Help.

To Check Out a Hijacked File

To keep the modifications in a hijacked file, check out the file:

1 In the right pane of the Snapshot View Update window, right-click a hijacked file.

2 On the shortcut menu, click **Check Out**.

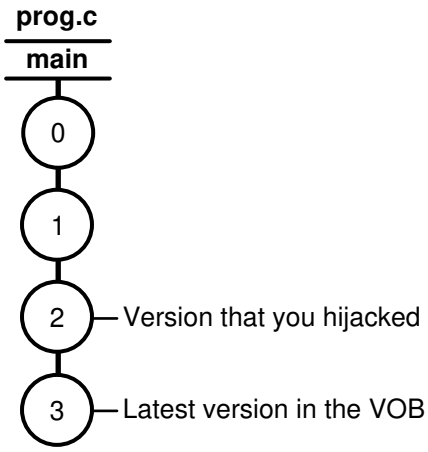
3 ClearCase treats a checked-out hijacked file as it does any other checkout.

When you are ready, you can check in the file.

Merging the Latest Version to a Hijacked File

If you are working with a shared set of versions and someone has checked in a newer version of the file while it was hijacked in your view (Figure 43), ClearCase prevents you from checking in the file.

Figure 43 Hijacked Version May Not Be the Latest Version



You have to merge the latest version in the VOB with the checked-out file before ClearCase allows the checkin.

When you issue the **Check In** command for a nonlatest version, ClearCase opens the dialog box to ask whether you want to merge the file now. If you choose to merge, ClearCase attempts to merge automatically, starting the Diff Merge tool if it needs your input to complete the merge. For information about using Diff Merge, see Help. After the merge, ClearCase prompts you to check in the file.

To Undo a Hijack

If, for specific hijacked files, you want to discard your changes and get a fresh copy of the version from the VOB, you can undo the hijack.

1 In the right pane of the Snapshot View Update window, select one or more hijacked files.

2 Right-click the selected files, and click **Undo Hijacked File**.

ClearCase overwrites the hijacked file with the version that was loaded in the view. If you want to overwrite hijacked files with the versions the config spec selects in the VOB, see Step 3 in *Updating the View* on page 158.

Under the Hood: Determining Whether a File Is Hijacked

To keep track of file modifications in a snapshot view, ClearCase stores the size and last-modified time stamp of a loaded file (as reported by the Windows file system). ClearCase updates these values each time you check out a file, check in a file, or load a new version into the view.

To determine whether a file is hijacked, ClearCase compares the current size and last-modified time stamp of a non-checked-out file with the size and time stamp recorded in the view database. If either value is different from the value in the view database, ClearCase considers the file hijacked.

Changing the read-only attribute of a non-checked-out file does not necessarily mean ClearCase considers the file hijacked.

Other Ways to Handle Hijacked Files

While updating the view, you can handle hijacked files in any of the following ways:

- Leave hijacked files in place
- Rename the hijacked files and load the version from the VOB
- Overwrite hijacked files with the version the config selects in the VOB

For more information, see *Updating the View*.

Updating the View

1. In ClearCase Explorer, select the snapshot view root directory.
2. Right-click to display the shortcut menu and click **Update View**.
3. To configure the Update Tool for handling hijacked files, in the Start Update dialog box click the **Advanced** tab and select a method for handling the remaining hijacked files. You have these choices:
   - Leave hijacked files in place
   - Rename the hijacked files and load the selected version from the VOB
   - Delete hijacked files and load the selected version from the VOB
4. To start the update, click **OK**.

Index

.keep files, canceled checkouts 86

A

activities
about 2
assigning in ClearQuest projects 69
basic tasks 75–94
changing assignment 92
changing ownership 93
checking out 76
ClearCase implementation details 15
ClearCase objects 67
ClearQuest objects 67
ClearQuest record types and 35
closing in ClearQuest 90
creating in ClearQuest 67
creating, methods for 66–68
deleting 92–93
delivering 99–118
delivering to project 10
dependences in deliver operation 105
duplicates 91
finding in ClearCase Explorer 62
finding, methods for 62–66
finishing 97
how to locate and create 61
measuring progress 89–94
moving to another development stream 92
postponing 91
reopening 91
setting from development view, methods for 70–72
type for integration 110
undelivered change sets 104
undelivered, in development stream 92
unsetting from development view 73
viewing change sets 85
working with 5, 8
adding files to source control 129–130

B

baselines 13
See also rebase operation
considerations in deliver operation 106
binary files, merging 112
building software
absolute paths in makefiles 139
ClearCase build tools 89
optimizing performance 46

C

case-sensitivity 53
ccase-home-dir directory ix
change requests
state types and transitions 40
change sets
comparing 84
undelivered 104
viewing 85
checking in
about 88

activities 100
comments, reusing 88
effect of on VOB links 80–81
procedure 88
checking out
about 76
directories 78
during deliver operation 114
during rebase operation 126
finding modified files 154
for remote use 152
hijacked files 156
nonloaded files 137
Open dialog box 77
when disconnected from network 154
checkouts
canceling 86
how cancellation is handled 86
ClearCase Explorer
about 25
navigation 28
starting 44
tool organization 26
ClearCase integrations, deactivating 152
clearexport and clearimport commands 133
ClearQuest
determining enabled for project 57
display of user database 32
logging on to user database 58
starting client 63
ClearQuest integration
assigning activities 69
closing activities 90
connecting to schema repository 57
creating activities 67
creating custom queries 64
deleting activities 94
finishing activities 97
modifying activity information 90
policies for deliver operation 102
queries 63
setting activities 71
setting up 57
UCM concepts 33
UCM schemas 34
comparing versions
hijacked files 156
procedures 83
components
about 14
choosing for snapshot view 52
loading into snapshot view 54
conventions, typographical ix
copying
nonloaded versions into views 137
snapshot views to removable storage devices 153
views from removable storage devices 155
cquest-home-dir directory ix
customer support xi

D

deleted files
canceling checkouts 88
deliver operation
about 10, 99
activity dependencies 105
baseline considerations 106
binary files 112
concurrent merge algorithm 109
how completed 116
merging activities 109

MultiSite method for 11, 117
posted 117
preparing work area 100
remote 117
starting 102
state model 116
stopping and restarting 113
turn off automatic merge 105
undelivered change sets 104
undoing changes 115
updating views 101
details pane 26
Developer Studio, deactivating ClearCase integrations 152
development streams
about 20
as parent streams 21
listing activities in 65
moving activities to 92
moving work from 141
naming convention 45
renaming 45
undelivered activities 92
development views
access to 54
adjusting scope of 137
delivering activities 100
preparing for rebase operation 120
removing versions from 126
setting activities in 70–72
unsetting an activity 73
development work areas
about 4
effect of rebase operation 123
preparing for deliver operation 100
testing after rebase 125
Diff Merge 110
directories
adding to source control 129
canceling checkouts 87
checking out 78
creating shared 139
finding checkouts from 84
how versioned 78
importing directory trees to source control 133
listing nonloaded files 136
remote use 150
removing element names 134
VOB link behavior 80
documentation
Help description x
online access 1
drives
for snapshot views 139
dynamic views
activating and deactivating VOBs 138
activating VOBs 52
assigning to drive 49
behavior of VOB links 79
build tools available for 114
starting work in 56
view storage directory location 50
when to use 46

E

elements
See also files; versions
history of changes 83
information display 30
moving and removing 133
nonloaded, accessing 136

orphaned 87
renaming 135

F

files
See also elements; versions
accessing 54
adding for new development task 132
adding to existing directory tree 129
adding to source control 129
deleted, canceling checkout 88
finding checked-out 84
listing nonloaded 136
VOB link 80–81
folder pane 26

G

get command 137

H

hard links 79
hardware configurations for remote use 150
Help, accessing x
hidden file 148
hijacked files
about 154
case-sensitive names 53
checking out 156
comparing to version on VOB 156
finding 154, 156
handling 155
how determined 158
merging 157
undoing hijack 157
History Browser 83

I

importing directories to source control 133
information pane 26
installing ClearCase vi
integration
deactivating ClearCase 152
UCM-ClearQuest 34
integration activity 110
integration streams
about 19
creating 45
evolution of 22
MultiSite issues 117
integration views
multiple-stream project 114
recommended view type for 47
removing versions from 116
updating for deliver operation 101
interoperation on Windows and UNIX 53, 145–147

J

Join Project Wizard, how to start 44
joining projects
about 43
creating streams 45
creating views 46

L

laptops
configuration for remote use 150
disconnecting from network 152
reconnecting to network 155
lsactivity command 65

M

makefiles, absolute paths in 139
merging files
binary files 112
during deliver operation 109
hijacked files 157
how it works 112
in rebase operation 124
resolving differences 110
turning off automatic 105
MultiSite, deliver operation and 11, 117
My Activities folder 29
MyToDoList query 38, 63

N

Network Attached Storage (NAS) device 51
Network Neighborhood, accessing views 140, 142
network operations, disabling 152

O

orphaned elements 87

P

parent streams 21
paths
absolute 139
version-extended 137
posted deliver operation 117
Project Explorer 65
project VOBs 15
projects
about 14
choosing components to load 52
creating activities 66–68
joining 43
loading components into view 54
moving work to other 141
multiple-stream 1
single-stream 2

Q

queries in ClearQuest
about 37, 63
customizing 64

R

read/write performance of remote storage devices 150
rebase operation
about 13, 119
effect on development work area 123
how to complete 127
interrupting 124
merging requirements 124

preparing development view 120
rules 122
single-stream project 125
starting 121
testing tasks 125
undoing changes 126
registering snapshot views, procedure 142–143
remote deliver operation 11, 117

S

schema
about UCM-ClearQuest 34
schemas
connecting to UCM repository 57
schemas, ClearQuest 34
shared directories 139
shortcut menus, deactivated 148
shortcut pane 25
snapshot views
See also updating snapshot views; working from a remote location
access to nonloaded elements 136, 148
assigning to drive 139
behavior of and operating system 145
checking out during deliver operation 115
choosing project components 52
constraints on location 47
copying nonloaded versions to 137
copying to removable storage devices 153
hardware configurations for remote use 150
improving performance 52
loading project components 54
location of storage directory 48
moving 144
rebasing 123
registering 142–143
tool for 155
transferring to laptop 152
updating for deliver operation 101
view.dat file 139, 148
when to use 46
storage devices, removable
disconnecting from network 153
performance of views copied to 150
reconnecting to network 155
streams
ClearCase implementation details 16
creating for project 45
subst command 140
symbolic links 79

T

time stamps
searching for modified files 154
setting in Update Tool 153
to-do list, adding activities 69
typographical conventions ix

U

UCM (Unified Change Management)
about 1
accessing development views 54
basic concepts 13
UCMCustomQuery 38
under the hood
activating VOBs 52
adding files to source control 130
canceling checkouts 86

checking out from snapshot views 115
ClearQuest UCM concepts 33
concurrent merge algorithm 109
hijacked files, how determined 158
integration activities and baselines 110
merging files and directories 112
snapshot view storage directory 47
UCM concepts 13
Update Tool
about 155
detecting hijacked files 154
handling hijacked files 158
setting time stamps 153
updating snapshot views
canceled directory checkout 87
directories 89
handling hijacked files 158
moving the view 144
remote use of view 152
time stamps of changed files 153
VOB links 80–81
URL access 29

V

version-creation comments
reuse at checkin 88
version-extended paths 137
versions
See also elements; files
comparing 83
copying nonloaded into views 137
moving to another activity 86
removing from development view 126
view storage directories
about 47
constraints on location 50
disk space required 47
location for dynamic views 50
location for remote use 151
moving 144
view.dat file
about 139
regenerating 148
views
See also development views; dynamic views; integration views; snapshot views
about 14
case-sensitivity 53
creating for project 46
creating on Windows and UNIX 145
startup 29
types of 46
Visual Basic, deactivating ClearCase integration 152
VOB links
checking in directories 89
VOB namespace
listing elements in 136
VOBs
activating and deactivating 138
project VOBs in UCM 15

W

Windows Explorer
accessing views 142
deactivating ClearCase integration 152
deliver operation 103–104
Windows User Profile 142
work areas
about 4
finding activities in 5

setting up 44
update of 13
workflow in UCM 2
working from a remote location
about 149
hardware configurations 150
removable storage devices 150
updating view 152