Rational Software Corporation®

RATIONAL® CLEARCASE LT® RELEASE NOTES
UNIX

VERSION: 2002.05.20

Rational®
the software development company

Scope

These release notes present information about the definition, delivery, and operation of Rational ClearCase LT configuration management software. Any functionality or characteristics described in this document referring to ClearCase applies to ClearCase LT as well.

Product Definition 1

ClearCase LT software for UNIX operating systems offers client/server solutions for configuration management software.

Before you install ClearCase LT software, read the section *ClearCase LT Hardware and Software Requirements* in this document.

What's New in ClearCase LT 2

This chapter summarizes significant new and changed features in Version 2002.05.20 of Rational ClearCase LT.

Summary of Changes in This Release

This version of ClearCase introduces the following new and changed features:

- *New UCM Features*
- *Enhanced ClearCase Web Interface Support*
- *Enhancements to UCM ClearCase-ClearQuest Integration*
- *Enhancements to Base ClearCase-ClearQuest Integration*
- *Enhancements to ClearCase Integrations with Third-Party Products*
- *Enhancements to Installation*
- *GUI Support for Date and Time Preservation*
- *Additional Support for Moving VOBS*
- *New View Database Format*
- *Changes to clearhistory and clearmrgman*
- *Other Enhancements*
- *Changes to ClearCase Commands*
- *Changes to Documentation*

New UCM Features

This section describes the main changes to UCM in this version of ClearCase.

Development Stream Hierarchies

In the basic UCM process, the integration stream is the project's only shared work area. You may want to create additional shared work areas for developers who work together on specific parts of the project. Now you can accomplish this by creating a hierarchy of development streams. For example, you can create a development stream and designate it as the shared work area for the developers working on a particular feature. Developers then create their own development streams and views under the feature-specific development stream. In other words, the developers join the project at the feature-specific development stream level rather than at the integration stream level. The developers deliver work to the feature-specific development stream and rebase their streams to recommended baselines in that development stream.

Interproject and Intraproject Deliver Operations

You can now deliver work from an integration stream or a development stream in one project to an integration stream or development stream in another project. In addition, you can deliver work from one development stream to another within a project. These features make it easier to manage related areas of development within a project and parallel releases of multiple projects.

Nondefault Deliver Target Policies

The default target for a deliver operation is the source stream's parent stream. You can also deliver work to nondefault target streams. Because any stream can now be the target of a deliver operation, this release adds several policies that you can set on streams to control what kind of deliver operations the target stream will accept. For example, you can set a policy to reject deliver operations that contain changes to components that are not in the target stream's configuration. You can set these policies at the project level so that the policy settings apply to all streams within the project, or you can set the policies on a per-stream basis.

Composite Baselines

A baseline selects one version of every element visible in a component. This release lets you create composite baselines. A *composite baseline* is a baseline that selects baselines from other components. By creating a composite baseline that selects baselines from every component in your project, you can use one baseline to represent the entire project.

After you create a composite baseline to represent the project, the next time you invoke the make baseline operation on the composite baseline, UCM performs the operation recursively. If a component that contributes to the composite baseline has changed since its latest baseline, UCM creates a new baseline in that component.

A composite baseline can select other composite baselines. This feature allows you to use a composite baseline to represent a system that consists of multiple projects. The system-level composite baseline can select project-level composite baselines.

Multiple-Component VOBs

You now have greater flexibility in storing components in VOBs. For example, you can now store multiple components in a VOB. Within a VOB, you can change an existing directory tree into a component. The component's root directory must be the VOB's root directory or one level beneath it. To store multiple components within a VOB, place the root directory for each component one level beneath the VOB's root directory.

Enhanced GUI Support

This version of ClearCase provides enhanced support for UCM-related GUIs, as follows:

- Project Explorer supports sorting of data, both within the Detail view and in all property sheets. In any display of columns of data, you can sort a column by clicking the column header.
- If a deliver or rebase is in progress, users can now get detailed information about the status of the operation in progress from the stream property sheet's **Get Status** button. In addition, the detailed list of items to be merged in the operation provides additional options from the shortcut menu.
- Additional functionality is available from the Add and Change baselines dialog boxes.

Support for Triggers on Additional UCM Operations

This release adds support for triggers to the following UCM commands.

- **chbl**
- **chfolder**
- **chproject**
- **chstream**
- **mkcomp**
- **mkfolder**
- **mkproject**
- **rmbl**
- **rmcomp**
- **rmfolder**
- **rmproject**
- **rmstream**
- **setplevel**

Other New Features

Other new features in this product release include:

- Ability to deliver selected baselines from a stream.
- Deliver from Stream Preview option. The Deliver from Stream Preview dialog box now gives you the option to pause the merge to allow you to perform any merge-related activities. If you select this option, the merge operation stops when it identifies the versions that require merging, checks out all versions, and performs all directory merges. You can then perform any operations related to the merge. When you are finished, continue the deliver operation.

Enhanced ClearCase Web Interface Support

This version of ClearCase provides a number of enhancements to the ClearCase Web interface.

UCM Support

This version of the ClearCase Web interface supports the UCM developer role. A developer can use the Web interface to join a project, create new activities, perform work in those activities, deliver work to an integration stream, and rebase a development stream to recommended baselines.

Web-Based Diff/Merge Capability

Performing the UCM rebase or deliver operation requires the ability to merge changes in files. The ClearCase DiffMerge tool has been implemented for this purpose within the Web interface context. The DiffMerge tool is called when you perform a rebase or deliver operation from within the Web interface; you can also run the Merge Manager directly from the Web interface toolbar.

Improved Workspace Management for Web Views

ClearCase now provides workspace management capabilities for Web views which are similar to those for standard ClearCase snapshot view functionality on a local copy of element versions.

Improved Control Over Session Timeout Length

A new configuration variable has been added to the ccweb.conf file to control the length of time that elapses before an unattended ClearCase Web session times out. The new variable is **–session_timeout** *age_in_seconds* where the minimum value for *age_in_seconds* is 600 (10 minutes) and the default value is 14400 (4 hours).

Interop Text Mode Supported in Web View Creation

It is now possible to specify **interop text mode** as a characteristic of a ClearCase Web view during the view creation process.

Web Server Component Installation Is Now Optional

Installing the ClearCase Web server components is now optional; previously these components had been installed on all systems.

To install the Web Server components, select **ClearCase Web Interface Server** from the list of components to select for installation. The Web Interface Server components require about 3 MB of disk space.

Changes to Web Interface Documentation

The online help for the ClearCase Web interface has been updated to reflect the enhanced functionality of this feature. In addition, the chapter *Configuring a Web Server for the ClearCase Interface* in the *Administrator's Guide* for Rational ClearCase LT contains updated information on configuring the ClearCase Web server.

Enhancements to UCM ClearCase-ClearQuest Integration

The UCM ClearCase-ClearQuest integration has been enhanced to support relationships between a single PVOB and multiple ClearQuest user databases. In previous versions of the software, all UCM ClearCase projects controlled by a single PVOB had to point to the same ClearQuest user database. Now different projects can point to different user databases, even if they are all controlled by a single PVOB.

This integration is now supported on HP-UX 11.11 and AIX.

Enhancements to Base ClearCase-ClearQuest Integration

This release adds a new version of the base ClearCase-ClearQuest integration, which allows you to associate versions of ClearCase elements with change requests in a ClearQuest user database.

The integration uses triggers on ClearCase commands to allow developers to associate versions with change requests. In previous releases, the integration used a Visual Basic trigger on Windows clients and a Perl trigger on UNIX clients. This release adds a new Perl library, including a trigger, that runs on Windows and UNIX. The new integration provides the following benefits:

- Support for UNIX and Windows platforms is now in a common code base.
- The integration is implemented as a set of object classes that provide modularity, extensibility, and reusability. The integration loads optional classes dynamically depending on the client environment. For example, if ClearQuest is not installed on a client, the integration does not load classes that support the ClearQuest Perl application programming interface (API).
- Performance has been improved by eliminating unnecessary communications operations.
- Windows clients no longer need Rational ClearQuest installed to access ClearQuest user databases. Like UNIX clients, Windows clients can now use the ClearQuest Web Interface to connect to ClearQuest.
- Because the integration supports the ClearQuest Perl API, ClearCase clients on UNIX can now, like Windows clients, access Oracle user databases on ClearQuest hosts that run on UNIX. This provides enhanced support for customization by administrators, because changeable data is isolated from other layers of the interface.
- The integration provides improved security, in that Clearquest user logon information, stored in the .cqparams file, can no longer be copied as a file between users. The session user logon name is now encrypted into the file and checked when it is used to make sure that only the original user is using the file.
- The integration supports the association of multiple file imports (by **clearfsimport**) with a single change request.

Note: The new integration provides a text-based user interface for developers who use the **cleartool** command-line interface and a **clearprompt** pop-up window user interface for developers who use one of the ClearCase GUIs such as ClearCase Explorer (on Windows) or **xclearcase** (on UNIX). The new integration does not provide a GUI for developers. For Windows client hosts that have ClearQuest installed, you may want to use the existing integration that uses a Visual Basic trigger to provide a GUI for developers.

Because the new trigger package, identified as V2 in the ClearQuest Integration Configuration GUI, completely replaces the functionality provided by the existing Perl trigger, V1, on UNIX, the existing Perl trigger is obsolete and will be removed from ClearCase in a future release. Therefore, we strongly recommend that you use the V2 Perl trigger on UNIX.

Enhancements to ClearCase Integrations with Third-Party Products

This version of ClearCase includes the following new or improved integrations with third-party products:

- Forte Developer (C++ IDE) Support
- Forte for Java

Sun Forte for C++ Support

Forte Developer V6.0 now supports a ClearCase integration. This integration provides a set of commands and tools that allow a software development team to share, control, and track files created with Forte Developer.

Using this integration, you can check out, modify, and check in your files without leaving the Forte Developer IDE or starting another application. You can also use this integration to access files under version control.

You can access commands and tools supported by the integration from the Forte Developer's Tools menu.

The following ClearCase commands and tools are available:

- **Add to Source Control**
- **Check In**
- **Check Out**
- **Compare**
- **Properties**
- **Source Control History**
- **Undo Check Out**
- **Version Tree**
- **Help**
- **xclearcase**

The first nine commands and tools give you all the capabilities you generally need. However, when you need advanced capabilities, the last command on this list, **xclearcase**, gives you access to the complete set of functions of the ClearCase system.

Forte for Java

This release introduces the integration of ClearCase with Forte for Java version 3.0.

Enhancements to Installation

This version of ClearCase includes the following enhancements to installation.

ClearCase LT Install Uses Rational Common Licensing Scheme

ClearCase LT installation has adopted the Rational Common Licensing scheme. You can now request a Rational Suite license or a ClearCase LT license and customize license use order in the license map file.

Server Install Now Allows Selection of VOB Schema 53 or 54

During 2002.05.20 server installation, you can choose to install support for either VOB schema 53 or 54 if you are installing the server on the following platforms:

- Solaris
- HP-UX
- AIX

If you are installing the server on Red Hat Linux, VOB schema 53 only is supported.

GUI Support for Date and Time Preservation

This release provides support for preserving the modification time of a file when it is added to source control, checked out, or checked in using the ClearCase GUI. The command-line interface, through the **cleartool checkout**, **checkin**, and **mkelem** subcommands, already provided this functionality with the **–ptime** option.

Additional Support for Moving VOBS

ClearCase now supports the following:

- Moving a VOB from Windows to UNIX
- Moving a VOB from UNIX to Windows

The **vob_sidwalk** and **vob_siddump** tools enable you to read and change Windows SIDs that are stored in schema 54 VOB databases. These tools are typically used for moving VOBs from one Windows NT or Windows 2000 domain to another or to an Active Directory domain, or for moving a VOB between a Windows host and a UNIX host.

For more information, see the **vob_sidwalk** reference page and the *Administrator's Guide* for Rational ClearCase.

New View Database Format

This version of ClearCase uses a new view database format.

Existing view databases will be reformatted the first time the view is used after this version of ClearCase is installed. You may also reformat the views manually using **cleartool reformatview**. We recommend that you reformat views manually if they have large databases. Doing so minimizes the chances that clients will experience view access time-outs during the reformatting.

When a view database is reformatted, its size grows by about 50%, which increases the size of the total view storage area by 10%.

Note: After a view is reformatted to the new ClearCase format, it cannot be reformatted to an earlier format.

Changes to clearhistory and clearmrgman

clearhistory is being reimplemented to be more consistent with the Windows version; for example, it will have the **Comment**, **Labels**, and **Attributes** boxes under the event log. The only Windows feature that will not be available is the **File > Open** dialog box.

clearmrgman is also being reimplemented, based on the new deliver/rebase GUI.

Other Enhancements

The following sections discuss additional enhancements made to this version of ClearCase.

chevent-Type Triggers Set to MODIFY_TYPE Now Fire Correctly

chevent type triggers now fire correctly when a trigger is set on MODIFY_TYPE. The **chevent** operation for element triggers is now a suboperation of the MODIFY_ELEM operation rather than the MODIFY_MD and MODIFY_DATA operations that it fell under previously.

Improved Shared Lock Memory Manager on HP-UX 11

This release supports an improved Lock Manager on HP-UX 11 platforms. The Lock Manager takes advantage of a shared memory implementation to provide greater scalability and better performance under stress. For more information, see the *Administrator's Guide* for Rational ClearCase.

clearfsimport Checks for VOB Locks During Operation

The **clearimport** command now checks for VOB locks before processing an element or version. If it detects a VOB lock, it shows a message indicating that the VOB is locked and that it will wait 60 seconds to retry. It remains in this state until either the VOB is unlocked or the user interrupts the import.

rmelem Use Now Restricted to VOB Owner or Privileged User

Beginning with this version of ClearCase, use of **rmelem** is restricted to the VOB owner or privileged user unless no versions have metadata (labels, attributes, or hyperlinks) and all branches were created by the user.

rmelem Triggers Fire When Symbolic Link Is Removed

Removing a VOB symbolic link with the **rmelem** command now causes all-element **rmelem** triggers to fire. This works only for all-element trigger types and does not work with per-instance triggers.

recoverview Now Dumps or Loads Very Large Views

The **view_server** now dumps and/or loads very large views. Previously it would fail to dump or load a view if the dump file exceeded 2 GB in size. The **view_server** now creates multiple dump files for very large views, each of which is limited to about 1 GB.

Improved Error Reporting for diffbl

Previously, **diffbl** displayed confusing error messages when it failed while comparing imported baselines, initial baselines, or baselines/streams based off imported or initial baselines, to other, similar baselines or streams. These messages described the failure as resulting from the fact that the baselines or streams involved did “not share a common ancestor (initial or imported) baseline”.

Now the error message describes the cause of the failure more accurately as resulting from the fact that the baselines/streams “are derived from different import baselines or the initial baseline”.

xcleardiff Can Now Compare Files with Names containing ’-dir’

xcleardiff can now compare files whose pathnames contain the "-dir" substring. Before, **xcleardiff** could not compare such files because the "-dir" string can be used as a keyword for a ClearCase directory comparison.

Improvements to mktrtype, mkeltype, and cptype

It is now possible to use the **cleartool mktrtype** and **cleartool cptype** commands for creating and copying trigger types that have locked obsolete types in the restriction list. It is also possible to use the **cleartool mkeltype** and **cleartool cptype** commands to create and copy element types whose super types are locked as obsolete.

Improvement to describe

Previously, the command **cleartool describe oid:***oid-number* returned an extra version ID string when the version was not selected by the view. In the current version of ClearCase, it returns exactly one version ID string.

Changes to ClearCase Commands

This section describes new commands, changes to existing commands, and commands that have been made obsolete in this release of the product.

New Commands

Table 1 lists new commands in this version of ClearCase LT. In addition, the **clearprompt** command is now available as part of the minimal developer installation.

Table 1 New Command in ClearCase LT Version 2002.05.20

Command	Description
vob_sidwalk/ vob_siddump	Read and change Windows SIDs that are stored in schema 54 VOB databases.

New Options and Arguments

Table 2 lists new options and arguments to commands in this version of Rational ClearCase LT

Table 2 New Options and Arguments to Existing ClearCase LT Commands

Command	New option/argument	Use
chbl	**–nrecurse**	Causes the changes specified by **–full** or **–incremental** to only apply to the composite and not its members
chmaster	**–obsolete_replica** *old-replica-selector*	Transfers mastership of all objects in the replica specified with *old-replica-selector*
chproject	**–dmodcomp** *component-selector* [,...]	Removes one or more modifiable components
	–spolicy *policy-keyword* [,...]	Allows the specified policy to be enabled or disabled by individual streams in the project
chstream	**–target** *stream-selector*	Sets the default deliver stream for an integration stream only
	–ntarget	Clears the default deliver stream setting
	–generate	Forces an update of the stream's copy of the config spec
	–policy *policy-keyword* [,...]	Enables the specified policy
	–npolicy *policy-keyword* [,...]	Disables the specified policy
	–recommended {*baseline-selector* {,...}} | **default**}	Sets the list of recommended baselines for the stream
	–nrecommended {*baseline-selector* {,...}} | **default**}	Clears the list of recommended baselines previously set in a stream

Table 2 New Options and Arguments to Existing ClearCase LT Commands (Continued)

Command	New option/argument	Use
clearmrgman	**–deliver** [**–to** *target-view-tag*]	Specifies a view attached to the deliver target stream
	–deliver [**–target** *stream-selector*]	Specifies a nondefault deliver target in the same or a different project
	–rebase [**–stream** *stream-selector*]	Specifies a stream to be used as the source for the UCM rebase operation
	–directory	For each directory, considers only the directory itself, not the directory or file versions or VOB symbolic links it catalogs
	–nrecurse	For each directory version, considers the file and directory versions within it, but does not descend into its subdirectories
	–follow	Causes VOB symbolic links to be traversed
	–noautomerge	Turns off automated merging of directories
	–query	Turns off automatic merging for nontrivial merges and prompts you to proceed with every change in the from-versions
	–qall	Turns off automated merging. Prompts you to determine whether you want to proceed with each change
	–file *mrgman-file*	Loads the specified merge manager file
clearprompt	**–n**	Allows use of **\n** to represent the newline character

Table 2 New Options and Arguments to Existing ClearCase LT Commands (Continued)

Command	New option/argument	Use
cleartool	**–status**	In interactive mode, returns the status of each cleartool subcommand executed.
describe	*replica-uuid-selector*	Displays the properties of a replica by specifing its OID
deliver	**–target** *stream_selector*	Specifies an alternate deliver target stream in the same or a different project
	–baseline *baseline-selector* ...	Specifies a list of baselines to deliver
diffbl	**–baselines**	Displays differences in terms of member baselines for the named composite baselines
	–nrecurse	For composite baselines, displays differences between the named baselines themselves and not between their members
lsactivity	**–contrib**	Displays contributing activities delivered in an integration activity
lsbl	**–cview**	Lists baselines in the stream attached to the current view
lsvob	**–family** *vob-family-uuid*	Lists the VOB with the specified VOB family UUID
mkbl	**–adepends_on**	Adds dependency references to baselines in its member components
	–ddepends_on	Drops dependency references to baselines in its member components
	–nact	Makes a baseline only in the specified component
	–component *component-selector*	Specifies the component whose dependency relationship you wish to change. When used with **–import**, specifies the component that contains the label type

Table 2 New Options and Arguments to Existing ClearCase LT Commands (Continued)

Command	New option/argument	Use
mkcomp	**–nroot**	Specifies the nonroot directory for components which only contain baselines
mklabel	**–follow**	Label the targets of VOB symbolic links
mkproject	**–policy** *policy-keyword* [,...]	Enables the specified policy
	–npolicy *policy-keyword* [,...]	Disables the specified policy
	–spolicy *policy-keyword* [,...]	Allows the specified policy to be enabled or disabled by individual streams in the project
mkstream	**–target** *stream-selector*	Sets the default deliver target for an integration stream only
	–policy *policy-keyword* [,...]	Enables the specified policy
	–npolicy *policy-keyword* [,...]	Disables the specified policy
	–in *stream-selector*	Selects the parent stream of the stream to be created
rebase	**–stream** *stream-selector*	Specifies the stream to be rebased

Obsolete Options and Arguments

The **-title** *title* option/argument pair, previously used by the ClearCase commands **chfolder**, **chproject**, **chstream**, **mkfolder**, **mkproject**, and **mkstream**, is obsolete at this release of the product.

In addition, the **–generate** *old-object-selector* option to the **rename** command is also obsolete.

Obsolete Commands

The **clearexport_ffile** command is obsolete; use **clearfsimport** instead.

Other Changes

Redefinition of element types (**mkeltype –replace**) is no longer allowed in replicated VOBs.

Changes to Documentation

The following major changes have been made to the documentation:

- Integration of UNIX and Windows material in a single *Command Reference*.
- There is significant reorganization of material in the following Rational ClearCase manuals: *Administrator's Guide* and the (online) *Command Reference*.

Unified *Command Reference* for UNIX and Windows

The *Command Reference* is now published in a single edition for both UNIX and Windows. At the beginning of each reference page are tables that show the platforms and ClearCase Product Family products to which the interface described applies.

Manual Reorganization

Large sections of material in the *Command Reference* have been moved to other manuals, and some material from other titles has been moved to the *Command Reference*.

- Certain build-related information that was formerly published in *other titles* is now published in the following reference pages:
  - **clearmake.options**
  - **makefile_aix**
  - **makefile_ccase**
  - **makefile_gnu**
  - **makefile_pmake**
  - **makefile_smake**
  - **makefile_sun**
- Some reference pages that do not describe any user interface—but rather ClearCase internals or concepts—have been removed from the *Command Reference* and the information has been incorporated into other titles.

  In particular, information from the following obsolete reference pages is now published in the *Administrator's Guide* for Rational ClearCase LT:
  - **abe**
  - **admin_server**
  - **albd_server**
  - **db_server**
  - **db_dumper/db_loader**
  - **errorlogs_ccase**
  - **license.db**
  - **lockmgr**

- **registry_ccase**
- **type_object**
- **view**
- **view_server**
- **vob**
- **vob_server**
- **vobrpc_server**

- A new reference page, **intro**, provides a broad orientation by organizing all ClearCase Product Family commands into various lists.

Changes to the *Command Reference*

In previous releases, restrictions for commands were in a section called *PERMISSIONS AND LOCKS*. In this release, the section is named *RESTRICTIONS*. The section that was previously named *Permissions Checking* is now called *Identities*. There is a new section named *Mastership*, which describes any restrictions applied when the command is run in a replicated VOB.

The **msdostext_mode** reference page now describes the options **–r** and **–f**. The **–r** option resets the line counters for elements of type **text_file** that have been changed to a binary type, and the **–f** option forces a recalculation of the line count of all VOB objects of type **text_file**.

ClearCase LT Hardware and Software Requirements

3

This section lists the basic platform, hardware, and software requirements for running ClearCase LT software.

Supported Platforms

ClearCase LT 2002.05.20 runs on the platforms listed in Table 3.

Table 3 Supported Platforms for ClearCase LT Version 2002.05.20

Hardware platform	Operating system
Solaris SPARC	Solaris 2.6, 7, 8 (2.6 32–bit only, others 32– and 64–bit)
HP 9000 Series 700 and Series 800 (32- and 64-bit support for all 11.*x* releases)	HP-UX 10.20, 10.20 ACE[1] (except diskless workstations), 11.0, 11.0 ACE[2] and 11.11 (11i version 1.0)
IBM RISC System/6000	AIX 4.3.2, 4.3.3, 5.1 (32–bit support)
IBM PC-compatibles	Red Hat Linux 7.0 (2.2.16–22 kernel) Red Hat Linux 7.1 (2.4.2–2 and 2.4.9 kernels) Red Hat Linux 7.2 (2.4.7 and 2.4.9 kernels)

1 Supported HP-UX ACE releases are July 1997, April 1998, June 1998, June 1999, and December 1999 Workstation ACE, and April 1998, June 1998, June 1999, and December 1999 Networking ACE.
2 Supported HP-UX ACE release is November 1999 ACE.

This release of ClearCase does not include support for the following architectures:

- Red Hat Linux 6.2

For more information about differences in features and functionality by platform, see the *Platform-Specific Guide* in online help. To access the platform guide, go to ClearCase help and click **Help Topics**.

Supported Platforms for ClearCase Web Servers and Web Interface

The following platforms support a ClearCase Web server:

- Solaris SPARC
- HP-UX
- AIX
- Red Hat Linux

All supported ClearCase platforms support the ClearCase Web interface. For details about the Web servers and Web browsers supported on different platforms, see *Basic Software Requirements* on page 24.

Supported File Systems

Table 4 lists the file systems that ClearCase LT supports for view and VOB storage. If a file system does not appear in the table, it is not supported. Inform Rational Technical Support or your sales representative of any concerns you have about this list.

For information on our support of NFS implementations, see *NFS Support* on page 22.

Table 4 Supported File Systems by Platform

Platform	Supported file systems
Solaris SPARC	UFS, VxFS (Veritas)
HP-UX	JFS, UFS, HFS, VxFS
AIX	JFS
Red Hat Linux	UFS, EXT2

The following file systems cannot be used to store any ClearCase LT data on any platform:

- Andrew File System (AFS), DCE Distributed File System (DFS)
- Any memory-based file system (such as memFS, tmpFS, swapFS)

NFS Support

Third-party automounters are not supported on any platform.

For a given platform, we support the NFS implementations that the platform supports.

If you use non-ClearCase access, see the *Administrator's Guide* for Rational ClearCase LT for a description of the limitations associated with use of NFS and potential workarounds.

Hardware Requirements

This section describes hardware requirements for installing and running ClearCase LT Server and ClearCase LT Client software.

Basic Hardware Requirements

For ClearCase client hosts:

- 64 MB main memory
- 300 MB disk space

For ClearCase server hosts:

- 128 MB memory
- 2 GB disk space
- 128 UNIX processes
- 600 UNIX file descriptors per host (not per process)

Note: UNIX processes and UNIX file descriptors per host are set as kernel parameters.

Disk Space Requirements for the Release Area

The file system of the networkwide *release host* must have sufficient disk space to hold the release area. The minimum disk space required for release areas on different platforms is shown below:

Solaris SPARC	209 MB	AIX	200 MB
HP-UX 10	217 MB	Red Hat Linux	119 MB
HP-UX 11	214 MB		

Disk Space Requirements for Individual Hosts

Table 5 shows the disk space requirement for each kind of installation. All the space must be contained in a single disk partition.

Table 5 Disk Space Requirements for ClearCase LT Release

Platform	Type of installation	Megabytes
Solaris SPARC	Full copy	175
	Standard	52
HP-UX 10	Full copy	184
	Standard	45
HP-UX 11	Full copy	182
	Standard	49
AIX	Full copy	240
	Standard	58
Red Hat Linux	Full copy	94
	Standard	17
All Platforms	Link	< 2[1] (install of ClearCase LT Client and Server)
	Mounted	< 2[1] (install of ClearCase LT Client and Server)

1 Disk space requirements for Link and Mounted installations represent the space required for items loaded in the **/var/adm/atria** directory.

In addition, any host that will have view directories needs enough disk space to contain all files loaded into the views and all view-private files added to the views. The amount of space required depends on the number and sizes of the files in the views.

The ClearCase LT server must have enough disk space to contain the files and databases used for storage of VOB- or view-storage directories. The amount of space required depends on the characteristics and use of the VOBs and views.

Software Requirements

This section describes software requirements for running ClearCase LT.

Basic Software Requirements

ClearCase LT requires the following software:

- The HTML Diff Merge (**xcleardiff**) tool requires a Netscape Web browser (version 4.6 or later, but before 6.0); the browser must support the level of HTML used in the files to be compared or merged.
- The Web interface requires
  - On the system acting as a ClearCase Web server, either an Apache Web server (version 1.3.14 and above), or an iPlanet Enterprise Server (version 4.1 SP8 through 6.0). The iPlanet server is supported on AIX systems only for version 4.1 SP8.
  - On any system accessing ClearCase through the Web interface, a Web browser, Netscape Communicator (version 4.72 through 4.78). Netscape 6 is not supported. (Note that it is not necessary to install ClearCase on such a system.)
- Use of the online manuals in PDF format requires an Adobe Acrobat Reader, version 3.0 or later.

Exporting the Release Area

Setting up the export is architecture specific; see Table 6. For details, see the standard reference pages for these files and programs.

Table 6 Exporting the ClearCase Product Family Release Area

Architecture	How to export ClearCase product family release area	
Solaris SPARC	1. Revise /etc/dfs/dfstab:	**share –F nfs /usr/ccase_rls** *options*
	2. Enter command:	**# /usr/sbin/shareall –F NFS**
HP-UX 10, HP-UX 11	1. Revise /etc/exports:	**/usr/ccase_rls** *options*
	2. Enter command:	**# /usr/sbin/exportfs –a** *other-options*
AIX	1. Revise /etc/exports:	**/usr/ccase_rls** *options*
	2. Enter command:	**# /usr/sbin/exportfs –a** *other-options*
Red Hat Linux	1. Revise /etc/exports:	**/ (rw)** - no command is necessary

Architecture Mnemonics and mount Commands

Table 7 provides the architecture mnemonic and sample CD-ROM **mount** commands for supported platforms. The architecture mnemonic is used as the name of root of the release area for each platform or set of platforms.

Table 7 Mounting the CD-ROM

Architecture	Mnemonic	Sample mount command
Solaris SPARC	**sun5**	Not necessary (if Volume Manager is in use)
HP-UX 10	**hp10_pa**	**mount –F cdfs –r /dev/dsk/c3d0s2 /cdrom**
HP-UX 11	**hp11_pa**[1]	**mount –F cdfs –r /dev/dsk/c3d0s2 /cdrom**
AIX	**aix4_power**	**mount –o ro –v cdrfs /dev/cd0 /cdrom**
Red Hat Linux	**rhat_x86**	**mount –r /dev/cdrom /mnt/cdrom**

1 When installing ClearCase on an individual host, you must use the appropriate release area for the architecture on your host. For example, if you have HP-UX 11 on your host, you must install from the **hp11_pa** release area.

Getting Started 4

This section contains information to familiarize you with installing, setting up, and using ClearCase LT software.

Installation and Licensing

This section contains an overview of information on installing and licensing ClearCase LT software.

Pointer to Installation and Licensing Information

Refer to the *Installation Guide* for ClearCase LT for installation information. It provides instructions on installing ClearCase LT Server and ClearCase LT Client software, lists software prerequisites, and contains information about removing old versions of Rational software and troubleshooting.

Installation Issues

This section discusses restrictions or defects that involve the installation of the ClearCase product. Take into account the information in this section before or during installation to be sure that ClearCase software or particular features are installed properly.

License Server Install Clarification

When you install the FLEXlm license server, you are prompted to enter the location to install Rational products. The installation text incorrectly suggests that this program installs ClearCase LT. It is important to understand that this program installs only the FLEXlm license server, not a specific product.

To install ClearCase LT after the license server is installed, you need to run the ClearCase LT install program, as described in the *Installation Guide*.

Note: The FLEXlm installation text incorrectly refers to the previous version of ClearCase LT as LT 1.0.

Installing ClearCase LT with Rational Suite Development Studio

If you obtained ClearCase LT as part of the Rational Suite Development Studio (RSDS) for UNIX package, install RSDS before installing ClearCase LT.

When you install ClearCase LT after installing RSDS, the first question prompts you for the name of the ClearCase LT license server host. Press RETURN to accept the default value; ClearCase LT will retrieve the appropriate Development Studio license key from the information provided in the RSDS install process.

Deinstalling ClearCase May Remove Web Interface Views

By default, views for Web interface users are created under the host data directory for ClearCase (/var/adm/atria). If ClearCase is deinstalled, the view directories are deleted, but the views remain registered. To avoid leaving entries for nonexistent views in the ClearCase registry, do one of the following:

- Remove any views created by Web interface users before you remove ClearCase from a Web server that is used to provide access to the ClearCase Web interface.
- Use the **–view_storage** option in the ccweb.conf file to designate some other location for Web interface views.

Installation of UCM Integration with ClearQuest

To use the UCM integration with Rational ClearQuest, take into account the following issues with the compatibility and version support of the following elements:

- ClearCase version (client and server)
- ClearQuest version
- UCM schema package
- ClearQuest metaschema feature level

Consider the following points:

- The feature level of the metaschema for ClearQuest 2001.03.00 database is 3. The feature level for ClearQuest 2001A.04.00 or 2002.05.20 database is 5.
- A ClearCase 2002.05.20 client requires either a ClearQuest 2001A.04.00 or a ClearQuest 2002.05.20 client, because the integration of UCM with ClearQuest uses new ClearQuest API calls.

As a rule, the UCM integration is supported on all platforms that are common to both ClearCase and ClearQuest clients. It is important to note that for any given release, ClearCase and ClearQuest may not support all of the same versions of a particular vendor's operating system. The UCM integration is therefore only supported on the

set of operating system versions that are supported by both ClearCase and ClearQuest. For example, ClearQuest doesn't support 64-bit versions of Solaris and HP/UX but ClearCase does. The integration, therefore, only supports the 32-bit versions.

Table 8 shows the compatibility of different releases of ClearCase and ClearQuest, the UCM package revision number, and the ClearQuest database feature level.

Table 8 Supported Integrations Between ClearCase (Using UCM) and ClearQuest

ClearCase release	ClearQuest releases	UCM package revision numbers	ClearQuest database feature levels
Release 4.1	2000.02.10, 2001.03.00	2.0	3
Release 4.2	2001A.04.00, 2001A.04.20, 2002.05.00	2.0, 3.0	3, 5
Release 4.2 + patch	2001A.04.00, 2001A.04.20, 2002.05.00	2.0, 3.0, 4.0	3, 5
Version 2002.05.20 [1]	2001A.04.00, 2001A.04.20, 2002.05.00	2.0, 3.0, 4.0	3, 5

1 The UCM integration with ClearQuest is not supported for AIX in this release of ClearCase.

To upgrade to Release 4.2 from 4.1 and continue to use your integration of UCM with ClearQuest, you must perform the first two steps. The last two steps are optional.

1. Install ClearQuest.
2. Install the ClearCase client (or both ClearQuest and ClearCase simultaneously).
3. Upgrade your UCM-enabled ClearQuest schema with the new version of the UnifiedChangeManagement package.
4. Upgrade your ClearQuest user database to the new version of the schema.

Rational ClearQuest Compatibility with ClearCase LT

ClearCase LT supports the option of integration with Rational ClearQuest. You can integrate the software in two different ways:

- If you install ClearCase LT with the Unified Change Management (UCM) option, integration with the version 2001A.04.00 or 2002.05.00 of ClearQuest is supported.

If you have an older version of ClearQuest, you must upgrade to one of these versions before configuring the integration.

- If you install ClearCase LT without the UCM option, you can integrate with ClearQuest Version 2001A.04.00 or later using the ClearCase-ClearQuest integration 1.0.

Note: Support the UCM/ClearQuest integration is not provided for AIX platforms in version 2002.05.20 of ClearCase.

Recovering from an Unsuccessful Installation

You may encounter problems running **install_release** on previously installed systems. At a certain point, **install_release** attempts to shut down the running CPF product software on the system. This is done by running the shutdown script. In Release 4.*x*, this script is *ccase-home-dir*/etc/atria_start.

There is a known problem with the shutdown script. When the installation is no longer intact, the script sometimes encounters an error and prevents **install_release** from completing the installation or deinstallation.

Workaround: Delete *ccase-home-dir*/etc/atria_start, which lets the installation proceed as if it did not need to stop a current installation.

Note: If an installation continues to fail because CPF software is running, reboot the computer.

Installation Stops if Standard ClearCase Already Installed on System

If you attempt to install ClearCase LT software on a system that has ClearCase software installed, the installation procedure stops. The procedure displays a message saying that it has detected an existing installation of standard ClearCase and that it is stopping prematurely because of encountered errors. ClearCase LT and ClearCase software cannot exist on the same system. After the ClearCase LT installation stops, deinstall the ClearCase component as follows:

1. Change directory to the ClearCase release area (not the ClearCase LT release area).
2. This release area is recorded in the file /var/adm/atria/host.dat. You must use the standard ClearCase release area to deinstall the installed version of ClearCase. The ClearCase LT release area cannot be used to deinstall ClearCase.
3. Run the **install_release** script.
4. Choose the **Local Deinstall** method.

5 Choose the **Local** model.

6 Select the ClearCase component for deinstallation.

Local Deinstall preserves a number of system files—such as rgy, rgy/backup, license.db, and exports.mvfs—in /var/tmp/Atria.preserve. Delete these files. You may also have to delete any files left in the /var/adm/atria tree.

After you deinstall the ClearCase component, you can restart the ClearCase LT installation.

Installation Stops if Installation Directory Not Empty

If you specify an installation directory that is not empty, the install utility halts the install process to avoid deleting directory contents accidentally, and displays an error message.

In the previous version of ClearCase LT, the installation deleted the directory contents without any warning.

ClearCase Client Computers Need Consistent Character Encoding

All ClearCase client computers that access a common set of VOBs and views must use a single common character encoding system. If all computers are not configured this way, ClearCase operations may fail or produce confusing or unreadable output.

For example, the Japanese SJIS and Japanese EUC encoding systems are available. They both represent Japanese characters but are incompatible. For this reason, you cannot mix SJIS and EUC in ClearCase clients.

Installing the ClearCase Integration with Forte for Java

Root privileges are required to sucessfully install the ClearCase Integration with Forte for Java on the sun5 platform.

Platform-Specific Information Relating to Installation

This section provides information that varies from platform to platform. The *Installation Guide* for Rational ClearCase LT specifies when you will need this information and defines the terms used in this section.

Operating System Vendor Web Sites

You can find up-to-date information on operating system patches at the vendor Web sites listed in Table 9.

Table 9 Web Sites of Operating System Vendors

Vendor	URL
Sun	sunsolve.sun.com
Hewlett-Packard	us-support.external.hp.com
IBM [1]	techsupport.services.ibm.com/server/fixes
Red Hat	www.redhat.com

1 IBM suggests using the **fixdist(1)** utility to download patches

Layered Software Packages

In some cases, correct ClearCase processing requires installation of a layered software package. Before installing ClearCase on a host, see Table 10 to determine whether you need to install any such packages.

Table 10 Layered Software Packages Required by ClearCase

Host type	Package name	Description
Solaris SPARC	SUNWsprot	SPARCompilers, bundled tools; includes default make.rules file
	SUNWbcp	Binary compatibility
	SUNWscpr	Source compatibility, root
	SUNWscpu	Source compatibility, user
	SUNWmfrun	Motif RunTime kit to run ClearCase GUIs
HP-UX 10	Upgrade. UPG-TLINK	Compatibility links for file system
	X Window System: X11R5 (or later)	Window system
HP-UX 11	Upgrade. UPG-TLINK	Compatibility links for file system
	X Window System: X11R6 (or later)	Window system
AIX	bos.net.nfs.client	NFS
Red Hat Linux	XFree	X Window System Kit

ClearCase LT Patches Incorporated in This Release

ClearCase LT 2002.05.20 incorporates ClearCase 5.0 Patch Stream patches through cc_lt_p2002.05.00–4.

Upgrading from a Previous Release

This section discusses issues involved in upgrading from a previous version of ClearCase LT.

The *Installation Guide* for Rational ClearCase LT provides information necessary to install ClearCase LT. Here is some general information to keep in mind about upgrading:

- Make sure that all views and VOBs are fully backed up. For information on backing up VOBs and views, see the *Administrator's Guide* for Rational ClearCase LT.
- You do not need to upgrade your license server or get new ClearCase licenses. Licenses work with any version of ClearCase product family software, and ClearCase Version 2002.05.20 hosts can use a ClearCase license server running a 4.*x* version of the software.
- Be sure that VOB and view servers are upgraded before you upgrade client hosts; Version 2002.05.20 clients cannot access VOBs or views on hosts that are running an earlier release of ClearCase.
- Make sure the /var/adm/atria directory is backed up.
- Before upgrading to a new ClearCase release, you must complete all deliver operations that are in progress.

See also *Installation Issues* on page 27.

Upgrading to the New Feature Level for UCM PVOBs

This version of ClearCase introduces a new feature level, FL3, to support some of its new functionality.

To enable Version 2002.05.00 in UCM environments, you first install Version 2002.05.00 on all systems in a specific order (for UCM, this order is different from the order required at previous releases of ClearCase and MultiSite), then raise the feature level of VOBs and PVOBs, and finally set the recommended baselines. Following is the procedure:

1 Install Version 2002.05.00 on ClearCase hosts in the following order:

a. The license server.

b. The registry server

c. VOB servers on all replicas

d. All clients (clients must be upgraded, or will be inoperable)

2. Raise all PVOBs and VOB families to FL3 see the reference page for **chflevel**.

3. Set the recommended baselines on integration streams that are in use; otherwise, existing development stream configurations may be unexpectedly changed when you perform a **rebase -recommend** operation.

For more information, see the ClearCase white paper on migrating to Version 2002.05.00 on the Rational Web site, www.rational.com/support/products/clearcase.jsp.

If servers are upgraded to Version 2002.05.20 before all client systems have beem upgraded, new VOBs must be created on servers that run ClearCase Release 4.*x*, because Release 4.*x* clients cannot communicate with VOBs created on systems that run the new release.

After you upgrade the feature level on your PVOBs, you must also set the recommended baselines on integration streams that are in use. If you do not, existing development stream configurations may be unexpectedly changed when you perform a **rebase -recommend** command.

For more information, see the ClearCase white paper on migrating to Version 2002.05.20 on the Rational Web site, www.rational.com/support/products/clearcase.jsp.

Evaluating This Release of ClearCase

To evaluate,Version 2002.05.00, you can install it on one or more test systems in your existing environment of servers, clients, views, and VOBs and configure the test systems to use your 4.*x* license server.

ClearQuest Compatibility with ClearCase

ClearCase LT Version 2002.05.20 can be integrated with Rational ClearQuest software in two different ways:

- If you are using ClearCase with the Unified Change Management (UCM) process, a UCM-ClearQuest integration is built in to this release; you can use it with Rational ClearQuest Version 2002.05.20.
- If you are using base ClearCase (that is, not using the UCM process), you can integrate with ClearQuest 2001A.04.00 or later using the ClearCase-ClearQuest integration 1.0.

Restrictions and Guidelines for Using ClearCase

5

This section provides restrictions on and guidelines for using Rational ClearCase Version 2002.05.20 software that are considered noteworthy. These are not considered defects because the behavior reported is not expected to change in a future release of the product.

Guidelines for Using UCM

The following sections provide guidelines for using UCM.

PVOB Feature Level Requirements

This version of ClearCase introduces a new feature level, Feature Level 3 (FL3). To support all of the new UCM functionality provided with the new release, a PVOB must be upgraded to the new feature level. Until a 2002.05.00 PVOB has been upgraded to FL3, it will support ClearCase Release 4.*x* clients as well as Version 2002.05.00 clients, but without providing support for all the new UCM functionality delivered with the new release.

Specifically, the following new features are not supported, even on 2002.05.00 clients accessing 2002.05.00 servers, until the feature level of the associated PVOB has been raised to FL3:

- Creation of baseline dependencies (composite baselines)
- Creation of components without root directories
- Using composite baselines to configure a stream
- Creation of sub-VOB components
- Creation of nested streams
- Support for relationships between a single PVOB and multiple ClearQuest user databases

Installing this software on a PVOB server that is running a previous release of ClearCase does not automatically change the PVOB's feature level; you must raise the feature level manually. Before raising the feature level, all ClearCase clients accessing

that PVOB must be upgraded to Version 2002.05.00. ClearCase UCM operations will fail on clients that run previous versions of ClearCase if they try to access a PVOB whose feature level has been upgraded to FL3.

When Setting Up a Project, Make Components Read-Only Initially

This release of ClearCase supports the ability to change a stream's configuration from one using a modifiable component to one using a nonmodifiable one. However, we generally recommend that components initially be specified to be nonmodifiable when creating a new project. After verifying that the project builds and tests correctly, update the project policies to allow modifications to any or all components.

Notes on Using the UCM-ClearQuest Integration

The UCM process in ClearCase is enhanced for sites that have installed Rational ClearQuest by a very tight integration between the activity management provided by UCM and the change request management provided by ClearQuest. Use the following guidelines with the UCM-ClearQuest integration.

Transitioning an Activity Enabled for ClearQuest After Delivery

If the **Do ClearQuest action after delivery** policy is enabled on a UCM project, delivery of a UCM activity enabled for ClearQuest using the command-line interface (CLI) may result in an attempt to transition the activity to a Complete state type.

If the activity record has a field that must be filled in before it can transition to the Complete state, the program displays an error. An example is the Defect record type in the default UnifiedChangeManagement schema, whose Resolution field must be nonempty before it can be resolved.

Workaround: Modify the UCU_CQActAfterDeliver global script to include code similar to that below, which fills in the Resolution field when the activity is delivered.

```
REM Add complete resolution code
REM Defect record type requires Resolution field to be non-empty
'Get hook's session context
Set Session = GetSession()
'Get the entity
Set entity = Session.GetEntity(entity_type,entity_id)
REM If record type is "Defect"
... If(entity.GetEntityDefName = "Defect") Then
REM If Resolution field is empty
... If(entity.GetFieldValue("Resolution").GetValue = "") Then
REM Fill in required field
    session.EditEntity entity, "modify"
    Call entity.SetFieldValue("Resolution", "Fixed")
    msg = entity.Validate
REM Remember to do some action if validate fails
```

```
entity.Commit
End If End If
```

For information on editing entities, see the ClearQuest API documentation.

Schema Requires Submitted State

If you are applying the UCM package to a custom ClearQuest schema (as opposed to using the out-of-the-box Unified Change Management schema), be aware that this package depends on the existence of a state whose name is Submitted. If your custom schema does not include a Submitted state, you can apply the package to your schema by using one of the following methods:

- Before applying the package, temporarily rename the state that is the target of the Submit action to Submitted. After applying the UCM package, you may rename it to its original name.
- Create a dummy state called Submitted, and assign its state type to Complete. If you do this, you must also create a dummy action whose target is the Submitted state. After applying the UCM package, you may delete the dummy state and action.

Changes to UCM CustomQuery1 Do Not Appear in Dialog Box

When using the UCM-ClearQuest integration, the list of records displayed in the list on the Add To Source, Check Out, and Check In dialog boxes is generated by running the UCMCustomQuery1 query, which can be customized. (To see the effect of your changes, you must click **File > Save** to save the query edits.)

However, if you copied the Public Queries UCMCustomQuery1 query to your **Personal Queries** folder and edited it there, the changes are not immediately visible. To see your changes, you must stop the integration server process.

To do so on your UNIX clients, type **cqintsvr stop** on the command line.

Cannot Import UCM-Enabled Records from ClearQuest Database

In general, you cannot import UCM-enabled records from a ClearQuest database; ClearCase cannot guarantee that UCM information that references an arbitrary ClearQuest database is correct. However, this restriction does not prevent data recovery in the event of a data loss. Records may be successfully imported into a ClearQuest database if all of the following conditions are true:

- The records have been backed up from that database.
- The name of the database has not changed.
- The ClearCase UCM information in the record is still valid.

Cannot Delete a Project Record in ClearQuest

If you are working with a UCM project that is linked to a ClearQuest user database and attempt to delete the project record, you get a run-time error. You cannot delete the record or undo the CommitAction hook. The workaround is to use the **squid_patch** utility to force the **ucm_vob_object** field of the orphaned project to 0.

Availability of Check Mastership Policy

The ClearQuest page of the UCM project Properties Browser has a check box for the policy **Check mastership before deliver**. This is supported with the UCM 3.0 and UCM 4.0 package revisions. If you are using the 2.0 package revision, the check box is unavailable.

Changing a Project Name Does Not Automatically Update ClearQuest

If information in ClearQuest records is changed from a ClearCase application, such as **cleartool** or the UCM Project Explorer, the ClearQuest display may not always reflect the actual contents of the database. To refresh the display, close and reopen the database from the **File** menu.

When Running Integration, Certain Operations Do Not Require ClearCase Installed

The following ClearQuest operations do not require ClearCase to be installed on the user's system when working with the ClearCase-ClearQuest integration:

- Submitting new CQ records (Defect, BaseCMActivity, or UCM_Project).
- Modifying an activity record if its headline field has not been changed.
- Modifying a UCM_Project record if its name field has not been changed.

Rebasing a Stream

A stream can be rebased to a baseline that meets the following criteria:

- The baseline is not from the stream that is being rebased.
- The baseline is labeled. Baselines created by deliver operations are not labeled by default. For information about how to change a baseline's labeling status, see the **chbl** reference page.

Additional rules apply to integration streams and development streams in selecting a baseline:

- An integration stream can be rebased only to a baseline created in another project or to an imported or initial baseline of that project.
- A development stream can be rebased to a baseline that meets one of the following criteria:
  - The baseline was created in its parent stream.
  - The baseline is in its parent stream's foundation.
  - The baseline was created in a stream other than its parent stream and is contained in its parent stream. A baseline is contained in another baseline if all changes in the first baseline are included in the second baseline.

Rebase is typically used to advance a stream's configuration, that is, to replace its current foundation baselines with more recent ones. However, under certain conditions, rebase can be used to revert a baseline, and to add or drop a component in a stream's configuration. It can also switch to a baseline that is neither an ancestor nor a descendant of the current foundation. The rules explained above are general rules for all rebase operations. You need to satisfy only the general rules when adding a component to a stream. When you advance, revert, drop, or switch a baseline, you need to satisfy the general rules and some additional ones.

- To advance a stream's configuration, the new baseline must contain the current foundation baseline.
- To revert or drop a baseline for a component in a stream, one of the following conditions must be met:
  - The component is nonmodifiable.
  - The component is modifiable but has not been modified in the stream, and the component is not in the configuration of any child streams.
- To switch to a baseline that is neither an ancestor nor a descendant of the current foundation, one of the following conditions must be met:
  - The component is nonmodifiable.
  - The component is modifiable but has not been modified in the stream, and the component is not in the configuration of any child streams.
  - The component has been modified but the new baseline contains the current foundation baseline, and the component is not in the configuration of any child streams.

These rules ensure that any changes made in a stream are not lost when the configuration changes.

Clear Recommended Baseline Setting if Baseline Is Removed

If a baseline is recommended for a stream, and then that baseline is removed, clear the recommended baseline setting in the stream by typing the command **chstream –nrecommended**. Then you can set new recommended baselines by using the command **chstream –recommended**.

Delivering Changes to a Read-Only Component

The deliver operation cannot allow changes to a read-only component. This situation can occur if someone changes the component modifiability in your project. The actions you take depend on whether you are doing an intraproject or an interproject deliver.

For an 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 changes in the other components depend 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. Determine 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 use **findmerge** (**findmerge -insert**) to selectively merge the changes 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 an 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.

Recommending a Baseline

You can recommend a baseline for a stream if the baseline is from the stream or the stream's foundation.

For a baseline not from the stream or the stream's foundation, the following rules apply:

- The baseline must be from the same project as the stream.
- The baseline must be contained in the stream, which means the baseline has been delivered to the stream, or the stream has rebased to the baseline or one of its descendants.
- The baseline must contain the current recommended baseline, which means it must be a descendant of the current recommended baseline.

You are not required to recommend a baseline for every component in the stream's configuration.

You can clear the list of recommended baselines. Note that doing this step alone will cause problems when existing development streams rebase to the recommended baselines. The rebase operation will attempt to drop all baselines in the development streams' configuration. This operation will probably fail or produce errors and is, therefore, not desirable.

You can choose to reset the recommended baselines to baselines from the stream or the stream's foundation with or without clearing the recommended list. This allows the stream to return to a known correct state after being changed inadvertently to a bad list.

Delivering Selected Activities to a Nondefault Target

Delivering selected activities to a nondefault target is subject to the following restrictions:

- If the full set of activities in the stream violates one of the policy settings, you will not be allowed to proceed with the selected activity delivery, even if the selected activities would not violate the policies.
- If the full set of activities in the stream contains changes from a foundation baseline, you will not be allowed to deliver selected activities from this stream, regardless of the deliver policy settings.
- If the set of activities you want to deliver contain versions in components that are not visible in the target stream or are not modifiable by the target stream, you will not be allowed to deliver the set of activities. Delivering an activity requires that you deliver all versions in all components in the change set. You can move the versions of the missing or nonmodifiable components into another activity.

Remote Deliver May Fail if Remote Site Has Not Synchronized

It is possible for a remote deliver to include changes to a component when the component is nonmodifiable in the project. This only happens when the remote site has not synchronized with the local site and does not know that the component has been made nonmodifiable. When the integrator attempts to resume the remote deliver, a check is made to ensure that no versions in nonmodifiable components will be changed. If there are such versions in the deliver, the operation fails and the user has to cancel the deliver operation.

Notes on Using the Base ClearCase-ClearQuest Integration

This section provides guidelines for using base ClearCase and ClearQuest together.

Support Policy on Modifying Perl Trigger Source Code

In this release, the integration includes a new trigger, which runs on Windows and UNIX platforms. The release includes the source code, in the form of Perl scripts, that the trigger uses. The Perl scripts include extensive comments that describe the purpose of their classes. For additional source code documentation, including an architectural overview and formatted class documentation, contact Rational Support. However, Rational Support cannot and will not support changes that you make to this source code.

Guidelines on Using the ClearCase Web Interface

The following restrictions and guidelines apply to using the ClearCase LT Web Interface.

Restrictions on the Web Interface in a UCM Environment

If you are using the Web interface in a UCM environment, you cannot use the Web interface to work effectively in projects enabled for ClearQuest. You can view such projects, but you cannot perform any integration operations through the Web interface; also, you cannot use the interface as a suitable client in a UCM project enabled for ClearQuest.

Interactive Triggers Fail When Accessed Through Web Interface

The ClearCase Web interface supports noninteractive triggers. Interactive triggers, such as those that attempt to read input or create a window, fail.

If a trigger attempts to read input using **clearprompt**, the ClearCase Web interface prints this error:

```
clearprompt is not supported in the Web interface
```

If a trigger attempts to read directly from standard input, it receives an error, because standard input does not specify a valid file descriptor.

In addition, any trigger failure in the Web interface context displays this error message:

```
Interactive triggers are not supported in the Web interface.  If the
trigger was interactive, it may have failed for that reason.
```

Trigger script writers can detect whether a trigger is running in the Web interface context by checking for the environment variable ATRIA_WEB_GUI. It is set to **1** if you are running in the Web interface context.

Note that the base ClearCase-ClearQuest integration is dependent on the operation of interactive triggers; for this reason, the ClearCase Web interface is not a viable interface for using that integration.

Host Name Resolution Must Be Enabled on Web Interface Clients

The Java program used in the Web interface attempts to connect to the Web server to transfer files. Web browsers only allow Java programs to open connections to the server from which the programs were downloaded.

To enforce this rule, the Web browser on the Web interface client must be able to resolve the Web server's host name to an IP address. If you use a host name in a URL that cannot be resolved by the client host, the Java program cannot connect to the server. In this case, Web-interface file-transfer operations such as checkout, checkin, and download fail.

If the Web server is being accessed through a firewall by means of a proxy server, the proxy server being used must support DNS lookup outside the firewall.

ClearCase Web Interface Problem in Overriding Primary Group

When the ClearCase Web server on Windows logs on to a client, it sets the primary group to the designated primary group in the client user's domain account. In Release 4.0, you could not override this group setting. As a result, sites that use domain mapping to allow user accounts in multiple domains to share VOBs could not access those VOBs through the ClearCase Web interface.

Workaround: Specify a configuration variable in the ccweb.conf file, and add a value to the registry that enables domain mapping.

Adding the Variable to the ccweb.conf file

To enable a single Web server to support one primary group override, add the **–primary_group** variable with a *groupname* value to the ccweb.conf file. The allowable values for *groupname* are the same as for the CLEARCASE_PRIMARY_GROUP environment variable. The ccweb.conf file must be located in /var/adm/atria/config. If you need more than one primary group override, configure additional Web servers.

Setting the Registry

Typically, when domain mapping is used to allow users from multiple domains to access the same VOB, each user must create the DomainMappingEnabled value (set to **1**) in the HKEY_CURRENT_USER\Software\Atria\ClearCase\CurrentVersion registry key.

To enable domain mapping for a Web server, create the DomainMappingEnabled value in the HKEY_LOCAL_MACHINES\Software\Atria\ClearCase\CurrentVersion key on the Web server machine. The value must be of type DWORD and set to **1**.

If you log on directly to the computer instead of logging on through the Web interface, user values for DomainMappingEnabled override the machine value.

Netscape MOZILLA_HOME Environment Variable Must Be Set

When you use the ClearCase Web interface on a Netscape browser, the MOZILLA_HOME environment variable must be set to the Netscape Communicator installation directory. Otherwise, messages similar to the following may be displayed when you try to check out or download files.

```
Netscape:Error
Java reported the following error on startup:
java.lang.SecurityException: system classes were not signed.
Netscape: Error
# Error: Issuer certificate is invalid. (-8156)
# jar file:   ./java/classes/java40.jar
# path:       ./java/classes/java40.jar
```

We recommend that you check the Netscape Web site, www.netscape.com, for more information on general Netscape requirements.

Problems with Toolbar Menus When Using Netscape

Toolbar popup menus don't close properly when running the Web interface using Netscape Navigator. Rather than closing automatically when the pointer is moved out of the menu location, they close only when you reselect the menu button.

Some Fonts in Netscape on UNIX Are Difficult to Read

The fonts used for ClearCase Web interface pages in UNIX versions of Netscape Navigator can be hard to read. Users can work around this by setting font preferences in Navigator:

1. Click **Edit > Preference > Fonts**.
2. Set variable width font to **Helvetica (Adobe)**, size **12**
3. Set fixed width font to; size **Clean (Schumacher)** size **12**
4. Select **Use my default fonts, overriding document-specified fonts**

Web Views Can Only Be Accessed Using the Web Interface

Web views can only be accessed from the Web interface; they cannot be used with native ClearCase interfaces (for example ClearCase Explorer or **cleartool**). Various native GUIs may allow selection of a Web view (for example as a deliver target), but any attempt to access a Web view using a native ClearCase interfaces will result in a warning or error message.

Preparing Old Web Views to Run with ClearCase 2002.05.20

Rational ClearCase Version 2002.05.20 introduces a new ClearCase Web interface. This interface is not compatible with Web views created by previous ClearCase Web interface clients. If you want to reuse existing Web views, we recommend that you clean them up before you upgrade the ClearCase Web server host to ClearCase 5.0.

Use the following procedure:

1. Check in all checked-out files and directories, or cancel the checkouts.
2. Make a note of the files and directories you have loaded into the view. You can use this information to configure the view to load these objects after the Web server host has been upgraded.
3. Unless the view contains important view-private files or directories (or hijacked files that you cannot convert to checkouts and then check in), delete all files and directories under the Web view root directory, but not the root directory itself.

Guidelines on Using the XML Diff Merge Tool

The tool presents some problems with scrolling:

- Windows restricts the height of a window to 32,767 pixels. This may not be enough to display the contents of large XML files. If this maximum size is exceeded, XML Diff Merge compensates for this restriction by reporting the number of lines that are not visible. This line count is displayed in the title bar of any affected tree view pane. The size (height) of the tree control window may be reduced by collapsing tree nodes. When nodes are collapsed or expanded, the count of lines not visible is updated. When the entire file is visible, the title bar indicator is removed.

  The collapse branch, collapse level and collapse attributes operations can be used to collapse many nodes in the tree at once. For navigating differences in an especially large file, we recommend that you collapse the tree entirely by selecting **collapse branch** on the root element, and then use the difference navigation buttons to expose the individual differences.

- If a tree control is scrolled horizontally, and then the control is widened such that the horizontal scroll bar is removed (the control is now wide enough to display the entire horizontal contents), the tree remains scrolled horizontally. Its scroll position should be pulled back to the left (zero).

  Workaround: Scroll the control to the left (zero) before resizing it.

- When an attribute name node is collapsed, it also displays the attribute value in the form *'name = "value"'*. The length of this summary text should be measured when computing horizontal scroll extents. Currently, *name* and *value* are measured separately, and this may cause the horizontal scroll extent to be too small if the *'name = "value"'* line is the longest line.

  Workaround: The right scrollbar button always scrolls, so any horizontal extent can be made visible.

- When the display is horizontally scrolled, the background rectangle of some attribute name nodes may not be computed correctly, causing a certain amount of the default selection rectangle to peek through. This default rectangle will be colored in the user's default selection background color.

  Workaround: Make the tree control wider, instead of using horizontal scrolling. The background rectangle will be displayed correctly.

Problems Using Netscape 4 to Compare HTML Files

A bug in some versions of Netscape Navigator 4 may cause problems when using Diff Merge to compare HTML files.

Under some circumstances, Netscape opens a mail window, instead of a browser window, when you try to render HTML files for comparison using Diff Merge. This can happen when you have both a mail window and a browser window minimized on the desktop. When you are comparing two HTML files and click **Render HTML**, the Netscape browser opens correctly. If you minimize the browser window and select the browser or another file to render, the mail window may open instead.

A different problem may occur if you close both the browser and mail windows and leave the Message Center open on the desktop. (The Message Center is a toolbar that can start, among other things, the browser and mailbox windows.) When you click **Render HTML**, Netscape attempts to open a new instance of Netscape rather than use the one that is running. As a result, you see multiple dialog boxes (some unreadable) from **xcompare** and a message from Netscape that it has found a lock file.

Guidelines on Using UNIX Snapshot Views

The following sections describe restrictions when using ClearCase snapshot views on UNIX platforms.

Version Tree Browser Opens a Temporary File

Using the Version Tree Browser to open a file from within a snapshot view on a UNIX system creates a temporary file that contains the text for that version of the element. Although the name assigned to the temporary file is not the version-extended pathname of the element, it provides all the information contained within that version-extended pathname, including the version number and branch structure of the selected element version. For example, the temporary file name for an element foo.c@@/main/11 would be *unique_id*_foo.c_main_11.

The temporary file is removed when the editor exits.

Version Tree Browser Displays Error on Attempt to Access Eclipsed Element

The Version Tree Browser now displays an error message if you try to access a checked-out version that is eclipsed. Previously, accessing the checked-out version would appear to work but the version actually accessed was the version visible in the view (that is, the eclipsing version) instead of the checked-out version. The error

message now includes the following text with the pathname of the checked-out version:

`Cannot_generate_name_for_checkout_in_view:<`*view_tag*`>`

Problems When Administrative VOBs Are Unavailable

An administrative VOB is used by one or more other VOBs as a central repository of global type objects. For a description of this feature, see the *Administrator's Guide* for Rational ClearCase.

ClearCase users may see errors when the administrative VOB is unavailable. Following are examples of situations when this may happen:

- A user attempts to attach a version label, using a label type that was previously created automatically, as a local copy of a global label type. The ClearCase **mklabel** command tries to contact the administrative VOB that contains the global label type. If that administrative VOB is unavailable, the **mklabel** command fails.
- A VOB backup script attempts to lock the entire VOB object of /vobs/proj before copying data to tape. For each administrative VOB used by /vobs/proj, the ClearCase **lock** command tries to contact the administrative VOB. If any administrative VOB is unavailable, the **lock** command fails, which causes the backup script to fail.

To disable the above checking for a particular ClearCase command (for example, to keep working while an administrative VOB is offline):

1. Become the **root** user.
2. Set the environment variable CG_PROCFLAGS to the value **no_process**.
3. Execute the command.

Restrictions on Use of VOB Schema 54

There are a number of restrictions on the use of the larger VOBs created using the extended VOB functionality (VOB schema 54):

- **reformatvob –rm** may not completely remove an old VOB database directory during the load phase of the reformat operation. If the reformat cannot be completed because there is not enough disk space on the host, remove the old VOB database directory manually. This directory has a name of the form

VOB-storage-directory/db.reformat. After you remove this directory, run **reformatvob –load**.

- **reformatvob** uses the **space** command to calculate the amount of space needed for a reformatting operation. This fails for large database files, although the failure does not cause the reformat itself to fail.
- The **space** command cannot successfully run the **stat()** routine on large database files. As a result, the **cleartool space –vob –generate** command can fail. This, in turn, can cause the **Standard ClearCase Daily Tasks** task, supported as part of new administration functionality in Release 4.0, to fail. If you have a VOB with large database files, the **space** command fails nightly on scheduled jobs.

  To avoid this problem, edit the scheduled job list so that it runs only on VOBs that do not have large database files.

If TZ Variable Is Set, ClearCase Uses It

If the TZ environment variable is set to a value different from the time maintained by the operating system, ClearCase uses the TZ time rather than the system time. In this case, file creation and change dates can be in error, and config specs may not work as expected.

Change to Behavior of Keep checked out Check Box

Prior to ClearCase Release 4.2, if you selected this check box, the view-private file that you added to source control would remain checked out. This behavior is consistent with that of the **cleartool mkelem** command. As a result, you could lose the contents of this file before it was truly part of the VOB. This was most likely to happen if you canceled the checkout.

In the current version of ClearCase, the file is checked in and checked out. You can continue working on the file, but its contents at element-creation time are preserved, even if you cancel the checkout.

Verbose Mode Slows xclearcase Performance

If you want **xclearcase** to display the following annotations, select **display version** in the browser preferences dialog box.

- `checked out but eclipsed`
- `eclipsed`
- `eclipsed by checkout`
- `checked out but removed`

- `error on reference`
- `view vob hard link`
- `no config record`
- `disputed checkout`

Note: Selecting **display version** slows **xclearcase** performance, especially if **xclearcase** is working with larger directories.

Problem Using dtpad Editor on Systems Running CDE

The **dtpad** editor that is part of the Common Desktop Environment (CDE) is implemented as a client/server application. By default, one **dtpad** server process is spawned for each **dtsession**, and all subsequent **dtpad** invocations run on clients that connect to this server. The server process, however, has no ClearCase view context, and thus cannot process VOB files properly.

Workaround: There are two possible workarounds for this problem:

- Invoke **dtpad** with the **–standalone** option. This forces the current invocation of the editor to run independently of the server process, and as such, it can run and retain the current view context.
- Before editing, start the **dtpad –server** process manually in a process set to a view. Subsequent invocations of **dtpad** then connect to this server. To edit files from a different view, stop and restart the server.

xclearcase Reuses Old Comments During Checkin

The checkin comment is taken from the comment text box. If the comment text box is blank, the new version is checked in with the comment string **“”** (empty string).

DDTS Integration Uses Wrong Environment Variable

The DDTS trigger scripts use the CLEARCASE_PNAME environment variable, but this environment variable is not set. Instead, the CLEARCASE_PN environment variable is set to the correct value.

The workaround is to set CLEARCASE_PNAME to CLEARCASE_PN at the beginning of each trigger that uses the environment variable.

Documentation Issues

This section presents late changes to documentation and describes errors or information missing from the documentation delivered with ClearCase software.

Problem with Reference Pages

The following problems relevant to ClearCase exist in the *Command Reference*.

config_ccase

The **config_ccase** reference page on UNIX systems does not mention the file /var/adm/atria/config/admin.conf, which allows or disallows remote administration of the host. Also, this reference page says that anyone can edit files in the ../config directory. That may not be true for all files there, including admin.conf. You must be **root** to edit admin.conf.

mount

The **mount** reference page should contain a section on *AUTOMATIC VOB DEACTIVATION AT SYSTEM SHUTDOWN* with the following contents:

At system shutdown, the architecture-specific ClearCase startup script is invoked with the **stop** option to execute the ClearCase shutdown procedure. As part of this procedure, a **umount -all** command deactivates all VOBs currently active on the local host.

On AIX 4, **umount -all** invokes the architecture-specific **mount** command **/sbin/helpers/mvfsmnthelp** with **U** as its first argument, and **/sbin/helpers/mvfsmnthelp** then invokes **umount(1M)**.

On other UNIX platforms, **umount -all** invokes the standard **umount(1M)** utility directly.

In addition, on some UNIX platforms, mounting VOBs with the **ro** option works properly; that is, it prevents writes to view-private files within the VOB, but does not prevent other clients from modifying the VOB nor does it prevent changes that do not go through the MVFS, such as some **cleartool** operations. However, on other UNIX platforms (namely, HP-UX 11.0 mounting VOBs with the **ro** option prevents view-private changes to the file namespace (such as creation or deletion of view-private files) but does not prevent writes to view-private files.

rebase

With the introduction of stream hierarchies at this release, most of the rules for rebasing an integration stream and a development stream are now the same. The **rebase** reference page contains sections on *Rules for Development Streams* and *Rules for Integration Streams*. These sections are superseded by information in the section *Rebasing a Stream* on page 38 in this version of the release notes.

Problem with *Developing Software* Documentation

The *Developing Software* documentation is missing information relating to the following topics.

Cannot Run ClearCase Commands in a UNIX Snapshot View from a Windows Client

You can access a snapshot view that was created on a UNIX host from a ClearCase client on Windows, but you cannot run ClearCase commands from that Windows client and have them be effective in the UNIX snapshot view. For example, if you use Windows Explorer to navigate to a team member's view on a different computer and right-click a file in the view, ClearCase options may not be available. To perform ClearCase operations on the files in a team member's view, register the view as follows:

1. In Windows Explorer, navigate to the root directory of the view you want to register.
2. Right-click the root directory of the view.

When you right-click the root directory of the view, ClearCase registers the view by adding an entry to your Windows User Profile.

Delivering Work to a Snapshot View

If you deliver your work to a snapshot view, you can encounter an error caused by your not having ever registered the snapshot view. This condition occurs because ClearCase cannot find the path to the snapshot view root directory until you register the view. Typically a view is registered when you run the **mkview** command.

On UNIX systems, if you are using the command line interface, you see the following messages:

```
Error: Can't find root directory of snapshot view with tag "..."
Error: Unable to determine view root of snapshot view "..."
Error: View "..." is inaccessible.
Error: Unable to prepare view common.
Error: Unable to deliver stream "..."
```

On UNIX systems, if you are using the GUI, you see the following messages:

```
It appears that you have never before accessed the snapshot view (view tag).
As a result, we're unable to locate the view's root path name
Please 'cd' to the snapshot view's root directory and do a
'cleartool update'. This establishes you as a user of the view.
```

If you see either of these messages, change directory to the view root directory of the snapshot view and use the following command in that directory:

cleartool update -print

This preview form of the **update** command is quicker than **update** and does not change the loaded files. After the command completes, retry the deliver.

Handling Elements That Are Not Visible

(The following material should be used as part of Section 5.4, *Merging Versions* in Part 2, *Working in UCM*.)

During a deliver (or rebase) operation, you can see the following warning message about elements not being visible in the integration view:

```
1 elements were skipped because they are not visible. You should
determine why they are not visible before you complete this deliver
or rebase operation. If these elements should be visible, cancel
this operation, fix the problem, and re-run the operation.
```

Do not ignore this situation. The deliver operation found versions of elements in the development stream that need to be considered for merging to the target stream, but did not find the elements in the target stream. Possible causes for this situation:

- A new element was added to source control, but the directory that catalogs the element is not checked in.

  In this case, cancel the deliver or rebase operation, check in the directory, and start deliver or rebase again.

- While a change to the element was being made (in the development stream, for a deliver; in the stream from which you are rebasing, for a rebase), someone operated on the element (in the target stream, for a deliver; in the development stream, for a rebase), as follows:

  - The name of the element or the name of a VOB symbolic link to the element was removed.
  - The element itself or the VOB symbolic link to the element was removed.

  In either case, decide whether the removal was appropriate. If the removal was appropriate, you can allow the deliver or rebase to ignore the element.

- The change may be obsolete, because you intended to remove the name of the element or the element itself.

Because ClearCase cannot tell what caused the situation, you must find the cause, fix the problem, cancel the current operation, and start over.

Problems with *Managing Software Projects* Manual

The wording of one of the UCM policies governing deliver operations has changed in this release to the following:

Do not allow deliver to proceed with checkouts in the development stream.

Previously, the wording for this policy was

Allow deliveries from stream with pending checkouts.

The online help and the *Managing Software Projects* manual continue to use the old wording for this policy.

Problems with Online Help

The following issues exist with online help.

Using xman to Display Man Pages

ClearCase and MultiSite reference pages are supplied in ASCII catman format in directories named cat1, cat4, and cat5. If you want to use **xman** to display ClearCase and MultiSite reference pages, you must create symbolic links named **man1**, **man4**, and **man5** in *ccase-home-dir*/doc/man that point to the cat directories. For example:

```
% cd /usr/atria/doc/man
% su
Password:xxxxxx
% ln -s cat1 man1
% ln -s cat4 man4
% ln -s cat5 man5
```

Caution: If you create man directories, do not run **cleartool man** as the **root** user. If you do, the man page will be deleted.

Issues with Bristol HyperHelp

The ClearCase online documentation is displayed using Bristol HyperHelp. If your site already is using HyperHelp, make sure that *ccase-home-dir*/bin appears in the path before any other reference to Bristol HyperHelp. Rational Software has extended HyperHelp to support special features in the ClearCase online documentation. The HyperHelp viewers supplied with ClearCase will display conventional HyperHelp files, but conventional HyperHelp viewers may not display ClearCase HyperHelp files.

Help Window Is Blank

Occasionally, a ClearCase help topic fails to appear: the title appears, but the body of the window is blank. When this happens, you can correct the problem as follows:

1. In the Help window menu bar, click **Options > Font > Normal**. The Help window is refreshed.
2. If the wrong help topic is displayed when the Help window is refreshed, click the **Back** button to return to the original topic.

Problems with API Reference Update and Release Notes for Rational ClearCase

The section Using the API on HP-UX states that the API examples include references to **-lperl** and **-lperlDynaloader**. The examples have been fixed for this release and no longer include these references.

Issues with Tutorials

This section describes problems with running tutorials.

Platform Support for Tutorials

Tutorials are supported for Solaris and HP-UX platforms only.

Status of ClearCase LT Software Change Requests

6

Noteworthy problems found in or resolved in Version 2002.05.20 of Rational ClearCase LT are listed in the file cc_issues.htm.

You can find this file in the directory *ccase-home-dir*/install/ after you've installed the product.

Note that any problems relating to installation or setup of ClearCase are noted in the section *Installation Issues* on page 27.

Contacting Rational Technical Support

7

If you have any problems with the software or documentation, please contact Rational Technical Support by telephone, fax, or electronic mail as described below.

For information regarding support hours, languages spoken, or other support information, click the **Technical 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