

Introducción a Rational ClearCase LT

Rosa María Carretero López
EPSA – Universidad de Castilla-La Mancha
rmcarretero@yahoo.es

1 Introducción

Rational ClearCase LT es un sistema de administración de configuraciones (CM: GESTIÓN DE CONFIGURACIONES) que maneja variantes múltiples de desarrollo de sistemas del software. ClearCase LT mantiene una historia completa de versión de todos los artefactos de desarrollo del software, código incluido, los requisitos, los modelos, los scripts, los activos experimentales, y las estructuras del directorio. Implementa políticas de desarrollo específicas, ofrece múltiples áreas de trabajo a desarrollar, y provee soporte adelantado para el desarrollo paralelo. ClearCase LT incluye Gestión unificada de cambio (UCM), un proceso opcional para organizar equipos de desarrollo del software y sus productos de trabajo.

Para convenciones tipográficas y documentación en línea ver anexo.

2 La visión general de ClearCase LT

Este punto introduce las características de Rational ClearCase LT y describe las diferentes maneras de usar el producto.

2.1 Beneficios de usar ClearCase LT

Rational ClearCase LT es un juego de herramientas extenso con el que se puede gestionar y seguir la pista de recursos del software, ya sean páginas Web, datos críticos para la misión, documentación, o código fuente.

Además ClearCase LT permite hacer lo siguiente con los recursos del software:

- Crear nuevas versiones de un recurso del software.
- Comparar versiones de un recurso del software.
- Combinar cambios de una versión en otra versión.
- Controlar cambios simultáneos para un recurso del software.
- Marcar ciertas versiones como fuentes estables para ser usadas.
- Determinar cuándo, cómo y por qué de un cambio en particular.

ClearCase LT también tiene algunas ventajas únicas:

- Soporte a los líderes de equipo que necesitan coordinar la actividad de las personas desarrollando productos conjuntamente.
- Da a los gestores del proyecto el control sobre la extensión y la frecuencia con la cual los miembros del equipo sincronizan su trabajo.
- Da soporte al desarrollo paralelo: Los desarrolladores trabajan en las áreas privadas y no afectan al trabajo de miembros del equipo.
- Ayuda a los integradores en combinar los esfuerzos del equipo de una manera controlada.

2.2 Terminología básica de ClearCase LT

Antes de empezar a describir las características más importantes de ClearCase LT, se deben tener en cuenta los siguientes términos:

- Un *elemento de archivo* es un archivo que contiene código fuente del software, un documento, código HTML, código XML, u otros datos que se almacenaron en un sistema de archivo.
- Un *elemento de directorio* contiene elementos del archivo y otros elementos del directorio.
- Una *versión* es una revisión específica de un elemento. Por ejemplo, en lugar de sobrescribir la misma copia de un borrador cada vez, lo que se hace es almacenar una copia de la primera versión, la segunda versión, etcétera (Figura 1).

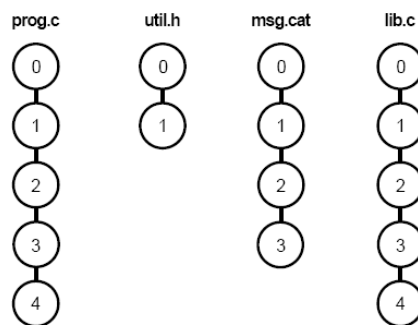


Figura 1. Elementos y sus versiones.

- Una *base objetos de versionado*, o *VOB*, es un repositorio que almacena versiones de elementos del archivo y elementos del directorio (Figura 2).

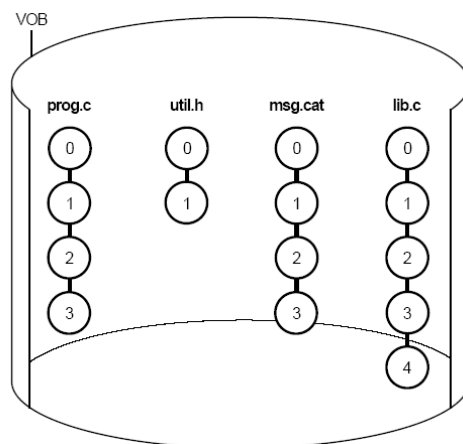


Figura 2. Almacenamiento de elementos en una Base de Objetos Versionados (VOB).

- Una *revisión fuera de la revisión de edición en el modelo* maneja cambios para el proyecto. Cuando se revisa un elemento, ClearCase LT crea una copia editable en la vista. Cuando se revisa un elemento, una nueva versión de él se agrega al VOB. Por ejemplo, si se revisa un capítulo de un libro de la documentación VOB. Se está trabajando en una copia del capítulo en la vista. Cuando se quiere que los cambios formen parte de la nueva versión del capítulo, se guardan los cambios y se registran en el capítulo (Figura 3).

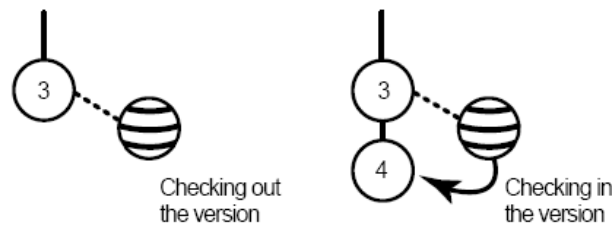


Figura 3. Usando la revisión fuera y en el modelo.

- Una *vista*, que es representada como un directorio, proporciona una vía de entrada para una versión específica de uno o más elementos en un VOB. Permite seleccionar un conjunto de versiones de elementos sin tener que especificar las versiones explícitamente. Proporciona un área de trabajo en el cual se puede trabajar en asignaciones retirados de otros desarrolladores (Figura 4). Por ejemplo, como se trabaja en la última versión de una página de Web, nadie puede ver los cambios que se han hecho hasta que se haya comprobado ese trabajo para el VOB.

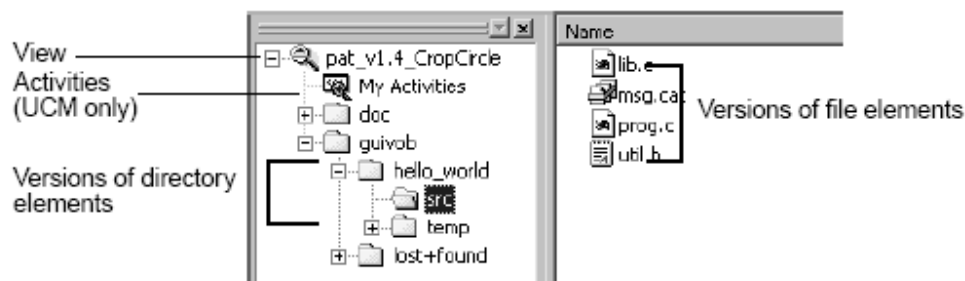


Figura 4. Uso de una vista para ver elementos compartidos.

2.3 Trabajando con ClearCase LT

La *gestión de configuración y de cambio* es un sistema o un proceso que asegura cada etapa de desarrollo de proyecto (planeando, editando, experimentando, construyendo, y desplegando) puede estar controlado y reproducido fácilmente.

ClearCase LT ofrece dos interfaces de administración de configuraciones:

- Gestión de Cambio Unificado (UCM), el cuál es un proceso basado en procesos de gestión de actividades de cambio.
- Base ClearCase LT, que es un juego de herramientas que se puede utilizar para crear una solución de gestión de configuración hecha a medida para un ambiente específico de desarrollo.

UCM está en capas sobre la base de ClearCase LT, lo que permite trabajar en UCM sin tener que dominar el conjunto básico de la herramienta. En UCM, el equipo de proyecto puede rastrear y manejar trabajo de proyecto de la actividad o nivel de tarea. El soporte de proceso para UCM se instala cuando es instalado ClearCase LT.

Cuando se usa la base ClearCase LT, el trabajo que se hace queda registrado en el archivo o en el nivel del elemento. ClearCase LT registra cuando cada archivo es creado o modificado. Los cambios de los archivos se convierten en una tarea de desarrollo, se concentran como si fuese un problema no resuelto en el software, no se guardan como un grupo. Para reconstruir el historial de una tarea de desarrollo, se debe examinar el historial individual de los archivos.

Se puede extender cualquier interfaz de administración de configuraciones integrándolo con Rational ClearQuest, una herramienta que rastrea defectos y peticiones de cambio.

Los siguientes puntos muestran una visión general del funcionamiento de ClearCase LT con y sin UCM.

3 Usar Gestión de Cambio Unificado (UCM)

UCM estructura el trabajo de un equipo de proyecto en un proceso definido y repetible. Este apartado muestra una visión general de los conceptos cruciales y las tareas involucradas al usar UCM.

3.1 Terminología UCM

Un *proyecto* es un producto específico de un determinado desarrollo, como puede ser un sitio Web corporativo o un proceso de cumplimiento de orden para comercio electrónico. En UCM, un *proyecto* es un objeto que contiene la información de configuración (por ejemplo, los componentes, las actividades, las políticas) necesita manejar y rastrear el trabajo en un producto. Un Proyecto UCM en ClearCase LT consta de un área de trabajo compartido y muchas áreas de trabajo privadas (normalmente una por desarrollador).

Un *componente* es cualquier grupo de código fuente y otros archivos o elementos pertinentes, algo semejante a un módulo de licencia o una GUI (Interfaz Gráfica Del Usuario) del cliente, que el equipo desarrolla, integra, y emite como una unidad. Los componentes constituyen partes de un proyecto, y los proyectos a menudo comparten componentes.

Una *actividad* es un objeto que registra el conjunto de archivos (*conjunto de cambios*) que un desarrollador crea o modifica para completar y entregar un tarea de desarrollo, como un problema no resuelto en el software (Figura 5). Una actividad consiste en varias versiones de dos o más elementos en una componente. Otras actividades pueden incluir una actualización para un archivo de ayuda en línea o la adición de una opción del menú para un componente de Interfaz Gráfica Del Usuario.

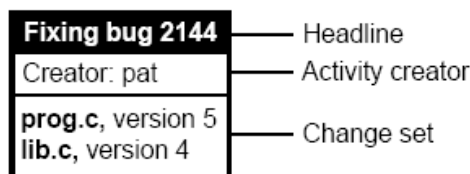


Figura 5. Una actividad

Un área de trabajo consta de una vista y una corriente. Una *vista* es un árbol del directorio que muestra una sola versión de cada archivo en su proyecto. Una *corriente* es un objeto ClearCase que mantiene una lista de actividades y líneas base y determina qué versiones de elementos aparecen en su vista (Figura 6).

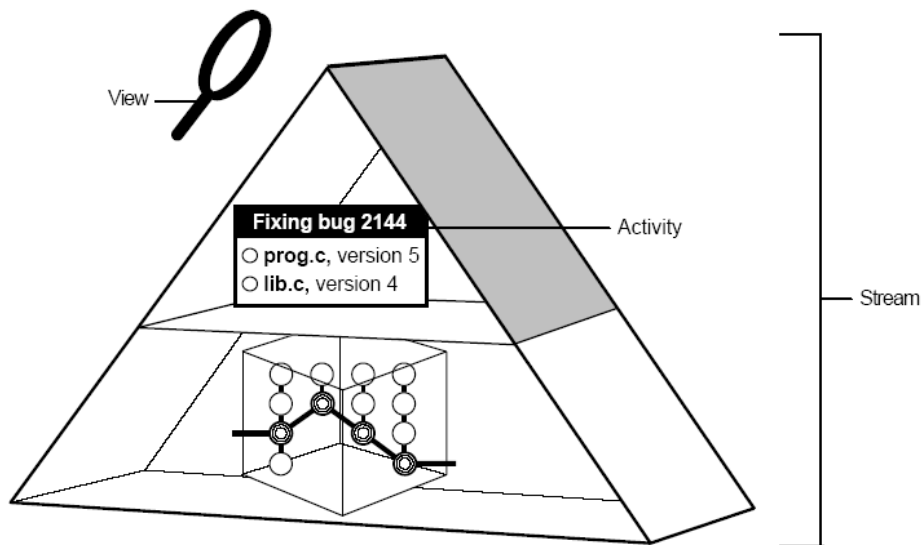


Figura 6. Una Corriente.

Un proyecto contiene una *corriente de integración*, cuya función es registrar las líneas base del proyecto y habilitar vías de entrada para las versiones de los elementos compartidos del proyecto. La corriente de integración y su correspondiente vista de integración representan el área de trabajo compartido del proyecto.

Cada desarrollador en un proyecto tiene un área de trabajo privada, que consta de: una *corriente de desarrollo* que mantiene una lista de las actividades del desarrollador y determina qué versiones de elementos aparecen en la vista del desarrollador y su correspondiente vista de desarrollo.

El trabajo de los miembros del equipo se integra en una componente, y cada versión de los elementos que la componen se identifican mediante una *línea base* (Figura 7). Representa una versión de un componente en una etapa particular en el desarrollo de proyecto, algo parecido a la primera parte redactada de un libro, un lanzamiento beta, o una versión del producto final. A lo largo de todo el ciclo del proyecto, el administrador del proyecto crea y recomienda líneas base y cambia el nivel de promoción de los atributos para reflejar hitos de proyecto.

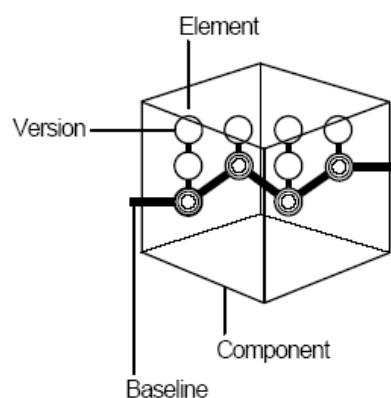


Figura 7. Una Línea Base (Baseline)

Cuando los desarrolladores unen el proyecto, incluyen en sus áreas de trabajo las versiones de directorio y los elementos del archivo representados por las líneas base recomendadas del proyecto. Alternativamente, los desarrolladores pueden unir el proyecto en un nivel específico en rasgo de corriente de desarrollo, en cuyo caso pueblan sus áreas de trabajo con las líneas base recomendadas en la corriente de desarrollo. Esta práctica asegura que todos los miembros del equipo empiezan con el mismo conjunto de archivos.

Si el equipo de proyecto trabaja con componentes múltiples, entonces se querrá usar una línea base compuesta. Una *línea base compuesta* es una línea base que selecciona líneas base en otros componentes. En la Figura 8, la línea base compuesta **ProjBL1** selecciona de líneas base **BL1** y **BL2**, cuyas componentes son **A** y **B**, respectivamente. El componente **Proj** no contiene ningún elemento propio. Su propósito exclusivo es contener la línea base compuesta que selecciona las líneas base recomendadas de los componentes del proyecto. Usando una línea base compuesta de esta manera, se puede identificar una línea base para representar el proyecto entero.

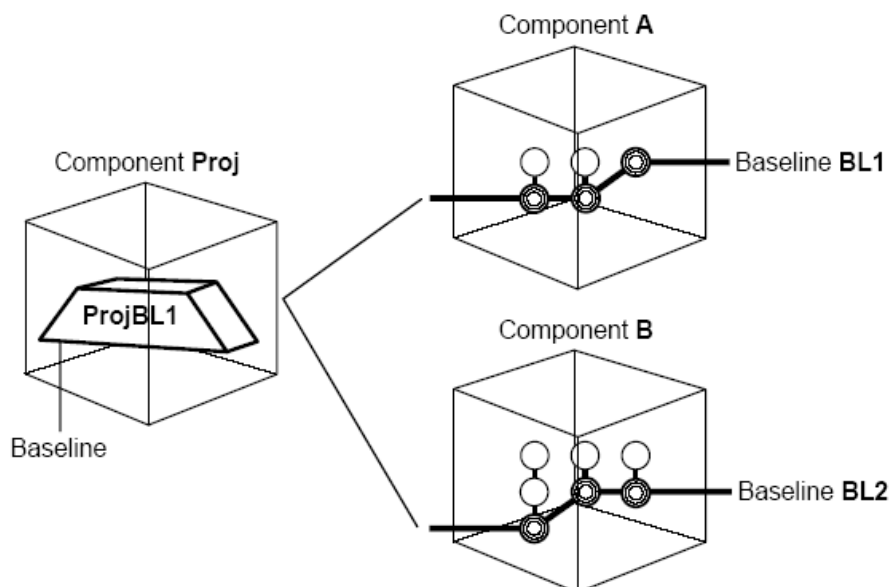
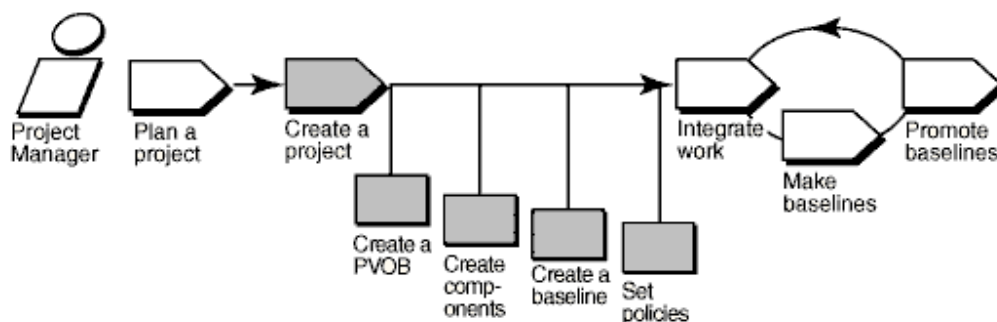


Figura 8. Líneas Base compuestas.

3.2 Flujo de trabajo de creación de un proyecto.

En ClearCase se tiene el siguiente ciclo en la gestión de un proyecto



Donde se tienen los siguientes pasos:

- El administrador del proyecto planifica el alcance del proyecto y toma decisiones sobre las políticas del mismo.
- El administrador de proyecto crea un proyecto, establece el área de trabajo compartido del mismo (la corriente de integración), e identifica un conjunto inicial de líneas base de uno o más componentes. Un componente es un grupo de elementos del directorio relacionado y del archivo, que son desarrollados, integrados, y emitidos conjuntamente. Una línea base es una versión de un componente.
- Después se hace una integración de todo lo construido durante el periodo de creación del proyecto.
- Periódicamente, el administrador del proyecto crea líneas base nuevas en el área de trabajo compartido que incluya el trabajo emitido.
- Del mismo modo, como la calidad y la estabilidad de líneas base se mejoran, el administrador de proyecto ajusta el nivel de promoción de líneas base para reflejar los hitos apropiados del proyecto.

3.3 Creando un proyecto en UCM

Antes de crear un proyecto, el gestor de proyecto mapea los archivos y directorios en la arquitectura del producto para hacer un conjunto de componentes UCM. Usando utilidades suministradas en ClearCase LT, el gestor o administrador del proyecto convierten en elementos los directorios y archivos existentes de sistemas del software de control de versión de terceros.

El gestor de proyecto crea un proyecto, junto con sus líneas base y de integración de flujo, mediante el explorador de proyecto. Se puede crear un proyecto nuevo desde la nada o se puede poblar un proyecto nuevo con líneas base de un proyecto existente.

NOTA: Cuando se instala el “ClearCase LT Server”, el “ClearCase LT Server Set Up” crea un *proyecto VOB* (PVOB), un VOB componente inicial, un proyecto inicial, y una línea base inicial.

El proyecto incluye un área de trabajo privado para cada desarrollador (Figura 9) y una área de trabajo compartido (Figura 10). El gestor de proyecto opera en el área de trabajo compartido.

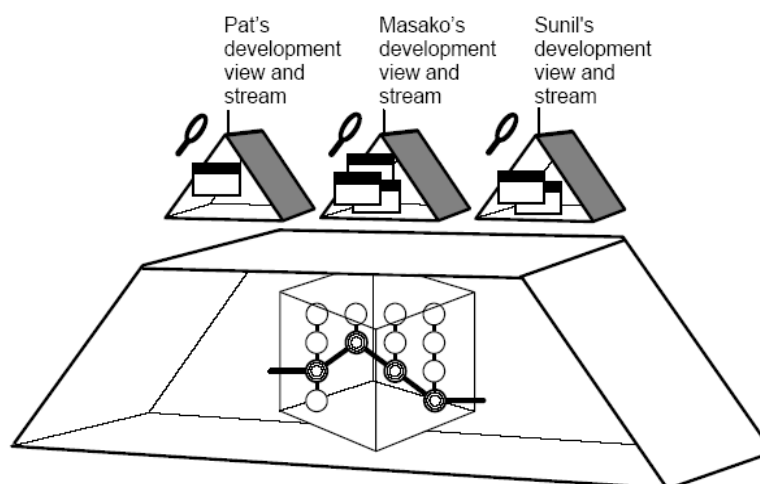


Figura 9. Áreas de trabajo privado.

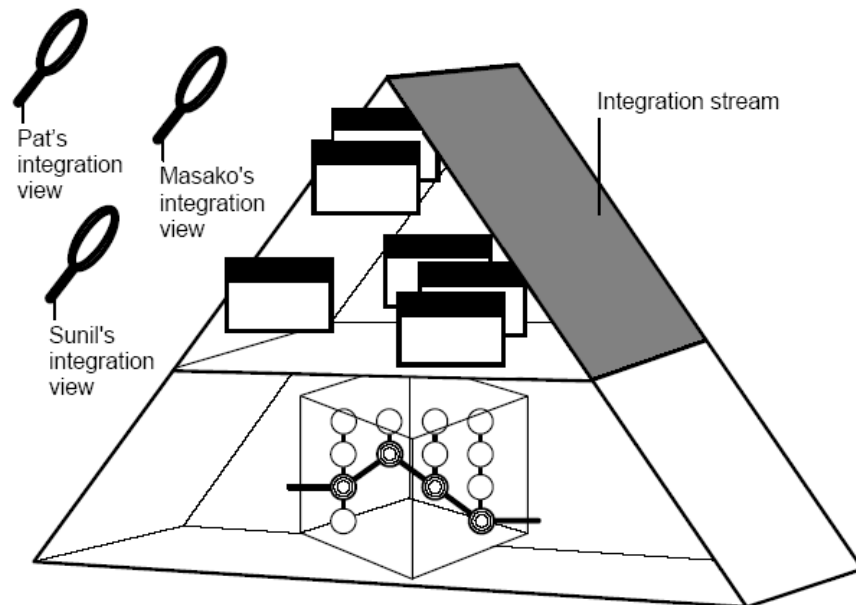


Figura 10. Área de trabajo compartido.

3.4 Uniendo el proyecto

Para unir el proyecto, se usa el “Join Project Wizard” para crear una *corriente de desarrollo*, una *vista de desarrollo*, y una *vista de integración*.

La corriente de desarrollo y la vista de desarrollo forman el área de trabajo privado, donde se trabaja sobre actividades. Se usan vistas de desarrollo para acceder y cambiar elementos. La corriente de desarrollo determina qué versiones de elementos aparecen en cada vista de desarrollo.

La vista de integración ofrece a los desarrolladores y a los gestores del proyecto acceso al trabajo que se entregó para el área de trabajo compartido.

3.5 Entregando trabajo al flujo de integración

Cuando el proyecto fue creado, el administrador de proyecto creó *un* área de trabajo compartido, o *corriente de integración*.

Cuando una persona está lista para integrar su trabajo con el trabajo del equipo, se *entrega* ese trabajo de la corriente de desarrollo a la corriente de integración. La operación de entrega puede llevarnos a combinar a combinar elementos si otro miembro del equipo ya ha dado versiones de uno o más de los mismos elementos (Figura 11).

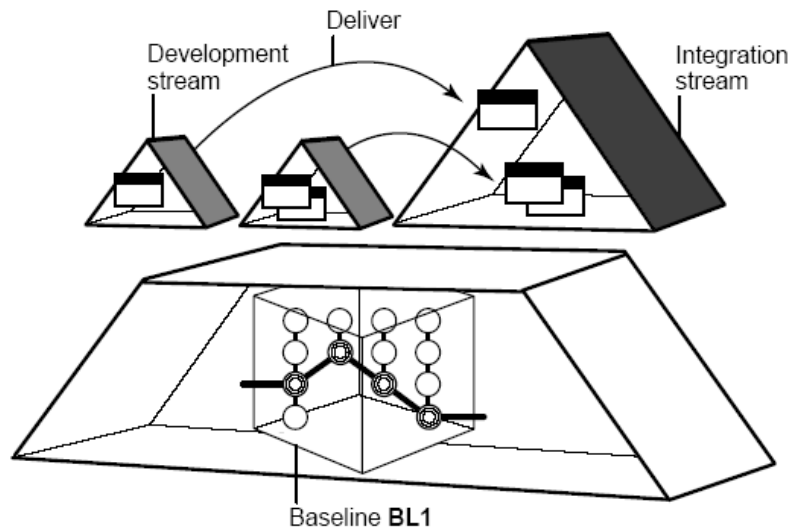


Figura 11. Entrega del trabajo al flujo de integración

Cuando el trabajo se ha entregado y ha sido combinado, se usará la vista de integración para probar el trabajo en la corriente de integración. La corriente de integración determina qué versiones de los elementos que han sido entregados deben aparecer en la vista de integración.

3.6 Rebasando áreas De Trabajo Privadas

Periódicamente, el administrador del proyecto crea una línea base nueva, lo cual incluye el trabajo entregado desde que la última línea base fue creada. Cuando esta línea base nueva pasa un cierto nivel de prueba, el administrador de proyecto lo organiza, designándolo como *la línea base recomendada* del proyecto.

Como se muestra en la Figura 12, se actualizará, o *rebasará*, la corriente de desarrollo para usar las líneas base nuevas recomendadas. La operación de rebase actualiza la vista de desarrollo a fin de que se incluyan las nuevas versiones de las líneas base. Si las líneas base incluyen las nuevas versiones del mismo elemento que fueron registradas en un área de trabajo privado (es decir, otro desarrollador ya presentó una nueva versión del mismo elemento para la integración la corriente), la operación de rebase fusiona las dos versiones en una nueva versión en la corriente de desarrollo.

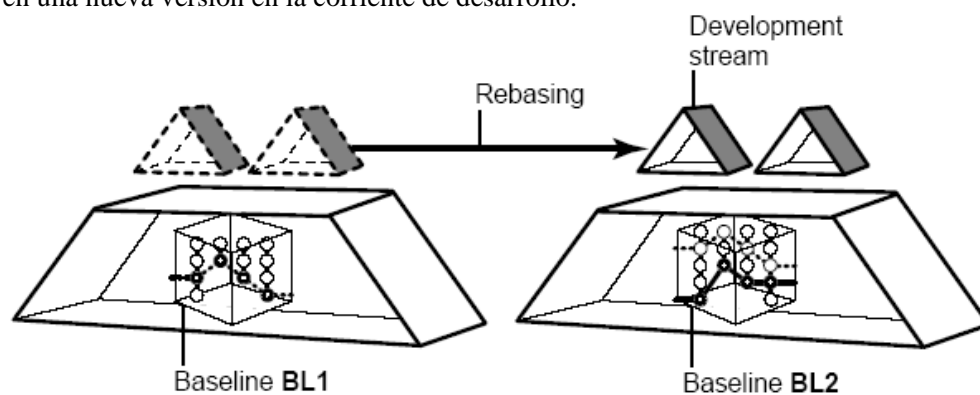


Figura 12. Rebasando el flujo de desarrollo.

3.7 Implementando políticas de desarrollo

UCM incluye un conjunto de políticas que se pueden usar en proyectos y flujos de desarrollo para imponer prácticas de desarrollo entre miembros del equipo del proyecto. Ajustando políticas, se puede mejorar la comunicación entre miembros del equipo de proyecto y se puede minimizar el problema que se puede encontrar cuando se esté integrando el trabajo. Por ejemplo, se puede ajustar una política que requiera que desarrolladores actualicen sus áreas de trabajo con la última línea base recomendada del proyecto antes de que entreguen el trabajo a la corriente de integración. Esta práctica reduce la probabilidad de que los desarrolladores necesiten trabajar en complejas combinaciones cuando entreguen su trabajo.

Además del conjunto de políticas que UCM provee, se pueden crear disparadores en operaciones UCM para implementar políticas de desarrollo hechas a medida. Un *disparador* es un monitor que especifica uno o más programas estándares o acciones incorporadas a ser ejecutadas siempre que una cierta operación ClearCase se realiza.

4 Integrando UCM con ClearQuest

“Rational ClearQuest” es una aplicación de gestión de petición de cambio. Usando ClearQuest, los miembros del equipo pueden proponer demandas múltiples de cambio para su producto, pueden mirar y pueden modificar demandas múltiples existentes de cambio, y crear consultas y preguntas específicas de ejecución para determinar el estado del proyecto.

La integración de UCM-ClearQuest aporta actividades de gestión adicionales para conectar actividades del proyecto en UCM con sus correspondientes registros en una base de datos de usuario ClearQuest. La integración deja al equipo del proyecto rastrear el progreso de una actividad desde la asignación hasta la terminación. Los administradores de proyecto pueden asignar las actividades a los desarrolladores, y los desarrolladores pueden usar las consultas de ClearQuest para encontrar sus asignaciones. Porque las actividades tienen registros correspondientes en una base de datos de usuario de ClearQuest, los administradores de proyecto puede usar todas las preguntas, la información, y hacer un mapa de servicios públicos en ClearQuest para ayudarles a gestionar proyectos.

5 Usando la Base ClearCase LT

Un equipo de proyecto puede usar herramientas ClearCase LT para desarrollar su propio sistema de gestión y rastreo de recursos software. Este punto describe los pasos básicos para establecer un ambiente de trabajo basado en ClearCase LT.

5.1 Terminología base de ClearCase LT

Una *rama(branch)* es un objeto que especifica una secuencia de versiones de un elemento. Cada elemento tiene una *rama central*, la cual representa la línea principal de desarrollo, y puede tener *subramas* múltiples, cada una de las cuales representa una línea separada de desarrollo (Figura 13).

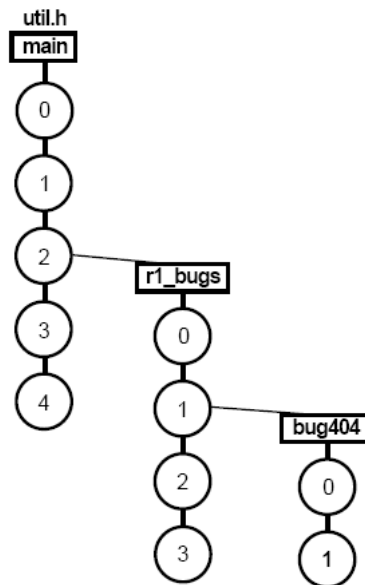


Figura 13. Ramas (branches).

Una *etiqueta de versión* puede estar vinculada a cualquier versión de un elemento para identificarla de una manera fácil de recordar. Una sola versión de un elemento puede tener varias etiquetas diferentes. Las etiquetas normalmente se aplican a un conjunto de elementos para marcar hitos importantes del proyecto o el punto de partida pensado para una rama (Figura 14).

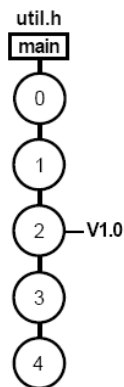


Figura 14. Etiqueta de versión.

Una *especificación de configuración* contiene las reglas usadas por una vista para seleccionar versiones de elementos. Las reglas son muy flexibles, y se pueden usar varios especificadores para indicar qué versiones se seleccionan. Por ejemplo, una especificación de configuración para una vista usada en el desarrollo en curso contendría reglas que elijan la última versión en la rama de desarrollo (Figura 15).

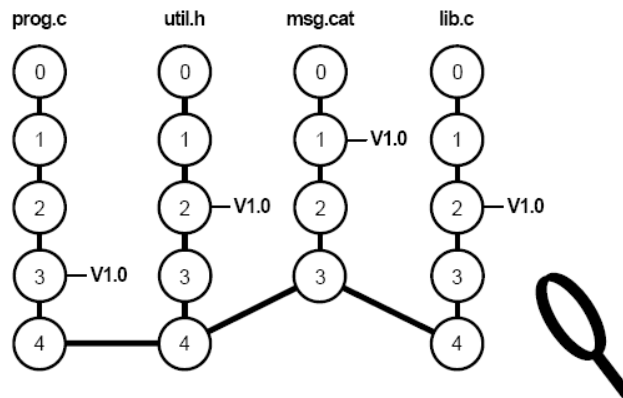


Figura 15. Selección de versiones usada en el trabajo de desarrollo.

Para examinar las versiones que fueron incluidas en una “release” particular, se usaría una especificación de configuración que usa una regla para seleccionar las versiones que fueron designadas para esa “release” (Figura 16).

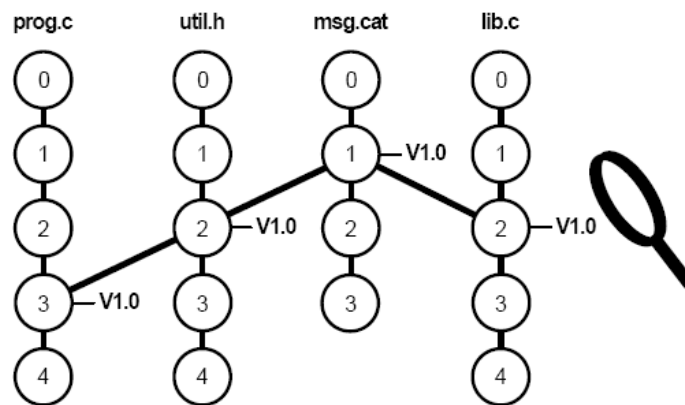


Figura 16. Selección de versiones incluidas en un “release” en particular.

5.2 Creación de un proyecto en base a ClearCase LT

Cuando los administradores de proyecto establecen proyectos en base a ClearCase LT, trazan un mapa de archivos y directorios en la arquitectura del producto para un conjunto de VOBs de ClearCase LT (igual que los administradores de proyecto hacen mapas de archivos y directorios para componentes ClearCase LT en UCM).

Después de que los archivos y los directorios son importados en VOBs, el administrador de proyecto normalmente etiqueta cada conjunto de versiones de elementos en esos VOBs para designar aquellas versiones en las que debería comenzar el desarrollo.

5.3 Planeando una estrategia de bifurcación (branching)

Cuando se usa la base de ClearCase LT, se deben usar directamente ramas para implementar desarrollo paralelo. (UCM usa flujos para gestionar las ramas)

ClearCase no depende de copias en el repositorio; aquí se crea una rama la primera vez que algún integrante de un grupo de desarrollo verifica un archivo fuera del desarrollo principal. Entonces, cualquier

miembro de ese equipo podrá trabajar con la rama de ese archivo. De este modo un porcentaje de los archivos se pueden ramificar en pequeños proyectos paralelos, haciendo que éstos sean mucho más fáciles de integrar con el trabajo de otros grupos. Este modelo disperso en ramas es mucho más eficiente que las copias en el repositorio.

En este punto hay un ejemplo simple de cómo definir una estrategia de bifurcación. Se debe tener en cuenta que la posible bifurcación y sus posibles combinaciones de unión son casi infinitas.

El equipo de proyecto puede usar muchas ramas concurrentemente. En la Figura 17, la **rama principal** representa la rama de integración, una subrama **r1_port** es un puerto para una plataforma nueva, una subrama **r2_beta** contiene trabajo para la versión beta de la siguiente “release”, etcétera.

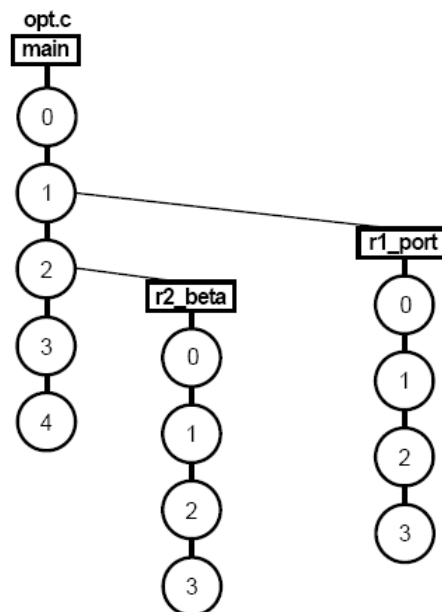


Figura 17. Ramas en un sistema de archivos de árbol de directorio.

5.4 Usando una rama privada

A veces, se puede querer crear una rama privada basada en la rama de proyecto. Por ejemplo, si se quiere trabajar en un problema no resuelto en el software sin afectar el trabajo que otros desarrolladores están haciendo. Para hacer esto, se debe escoger la versión en la que se quiere comenzar. Luego se puede crear una segunda especificación de configuración para una segunda vista que creará una rama privada comenzando en esa versión (Figura 18).

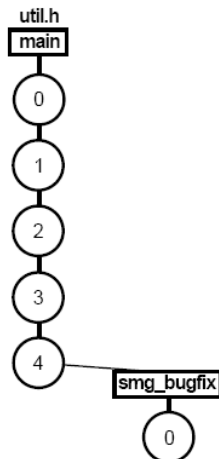


Figura 18. Trabajando sobre una rama privada.

Para incorporar el trabajo realizado en el proyecto, se deben incluir los cambios en la rama privada de la rama del proyecto. Si se decide abandonar dicho trabajo, no se deben incluir los cambios. En uno u otro caso, cuando se quiera reanudar el trabajo en la rama del proyecto, se debe cambiar la vista y volver a usar la configuración específica estándar del proyecto.

5.5 Combinando trabajo entre ramas

Para integrar trabajo de una rama en otra, se *inserta* el trabajo de una subrama a otra rama. Por ejemplo, en la Figura 19, todo trabajo debe ser combinado con la **rama principal** antes de que se incluya en una “release” del producto. Después de que la parte de la rama **bug102** se ha probado y aprobado para la integración, se combinará primero el trabajo para la rama **r1_fix**. En algún punto, se prueba todo el trabajo en la rama **r1_fix**; cuándo el trabajo esté incorporado en el proyecto principal, esta rama deberá insertarse en la **rama principal**.

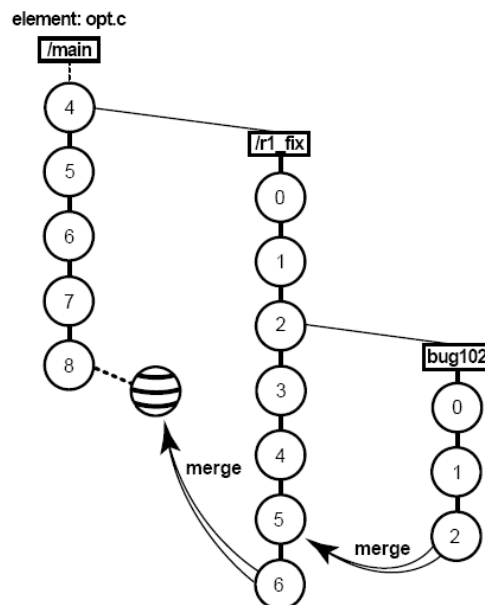


Figura 19. Combinando ramas.

5.6 Creando configuraciones específicas estándar

Para asegurar que los miembros del equipo están trabajando en las ramas correctas (directorios y versiones de archivos apropiadas), el administrador del proyecto crea una especificación de configuración estándar que los miembros del equipo usan en sus vistas.

5.7 Creando etiquetas

Una *etiqueta* en la base ClearCase LT es un nombre creado por el usuario que puede estar ligado a una versión. Las etiquetas son una herramienta potente para los administradores del proyecto. Aplicando etiquetas para grupos de elementos, se puede definir y conservar la relación de un conjunto de versiones de archivo y de directorio para cada punto dado en el ciclo de vida del desarrollo. Por ejemplo, se puede aplicar una etiqueta para todas las versiones consideradas estables después de la integración y prueba. Se debe usar esta etiqueta de la línea base como los cimientos del nuevo trabajo.

5.8 Usando meta datos para implementar políticas

En la base ClearCase LT, los administradores del proyecto usan *meta datos* para definir políticas de proyecto y anotar propósitos diferentes en el VOB. Los meta datos incluyen lo siguiente:

- Atributos.
- Tipos de rama.
- Hipervínculos.
- Cerrojos.
- Disparadores.
- Etiquetas de versión.

NOTA: También se pueden utilizar atributos, hipervínculos, cerrojos y disparadores cuando se usa UCM.

Para anotar objetos en el VOB, un administrador de proyecto puede usar atributos, hipervínculos, y etiquetas de versión:

- Los atributos son pares de nombre-valor que están vinculados a los objetos. Un uso para los atributos es poner números a las versiones; otro es identificar qué versiones en una línea base han sido probadas.
- Los hipervínculos conectan dos objetos VOB. Por ejemplo, se puede definir un hipervínculo para asociar una especificación de diseño a su ejecutable correspondiente.
- Las etiquetas identifican versiones específicas de elementos. Las etiquetas pueden ser usadas en las reglas de especificación de la configuración para identificar puntos bifurcados o marcar todas las versiones que fueron usadas en una construcción.

Para definir e implementar la política del proyecto, los administradores del proyecto pueden usar cerrojos y disparadores. Por ejemplo, una rama de integración puede estar bloqueada temporalmente para advertir a los desarrolladores de combinar el trabajo sobre ella. Un gestor de proyecto también puede crear un disparador que advierta a los desarrolladores para que incorporen el número de error asociado a una interrupción y agregar ese número a la versión.

6 Empezando a utilizar ClearCase LT

Este punto describe qué hacer después de instalar Rational ClearCase LT y dónde encontrar más información acerca de cómo usarlo.

6.1 Primeros pasos para gestores de proyecto y administradores

Después de que se haya instalado ClearCase LT Server en el ordenador del servidor y se haya instalado al cliente ClearCase LT Client en su ordenador, se necesita planear y crear estructuras de proyecto para los desarrolladores.

6.1.1 Usando UCM

El programa ClearCase LT Server Set UP crea un *proyecto VOB* (PVOB), un componente inicial VOB, un proyecto inicial, y una línea base inicial. En Windows, también pregunta si se quieren importar datos en ClearCase LT y ejecutar el Import Wizard, si es apropiado.

Después de que el programa Server Set Up termine, se puede iniciar desarrollo. También se pueden crear proyectos adicionales, componentes, y líneas base, pero no es necesario.

NOTA: Un proyecto debe contener uno o más componentes, y los proyectos pueden compartir componentes.

6.1.2 Usando la Base ClearCase LT

El programa ClearCase LT Server Set Up crea un VOB y un VOB administrativo. Un *VOB administrativo* es un VOB que contiene un tipo global de objetos que son copiados en los VOBs del cliente, en unas bases que son necesarias cuando los usuarios quieren crear instancias de ese tipo de objeto en los VOBs del cliente. En Windows, ClearCase LT Server Set Up también pregunta si se quieren importar datos en ClearCase LT y ejecutar el Import Wizard, si fuese necesario.

Después de que el programa Server Set Up termina, se pueden crear VOBs adicionales. Cada VOB que se crea es asociado con el VOB administrativo. Si se escoge no usar al VOB administrativo, entonces se pueden ignorar estos vínculos.

También se deben crear estos artículos adicionales:

- Los tipos de rama, para representar las diferentes líneas de desarrollo que hay en la organización.
- Las reglas de especificación de configuración, controlar las versiones que fueron seleccionadas por la vista.
- Tipos de etiquetas y etiquetas, para identificar el punto de partida de una rama o marcar hitos de proyecto.

7 Primeros pasos para desarrollar sobre Windows

Después de que instalar ClearCase LT en el ordenador, se debe hacer lo siguiente:

- Arrancar el explorador de Rational ClearCase LT.
- Completar el tutorial de ClearCase LT.
- Establecer áreas de trabajo e iniciar el trabajo.

Los siguientes puntos describen estas tareas.

7.1 Ejecutando el explorador Rational ClearCase LT

El racional ClearCase LT Explorer es la interfaz para su ambiente de desarrollo. Para iniciar el explorador de Rational ClearCase LT, se debe hacer doble clic sobre el icono **Rational ClearCase LT** del escritorio o seguir las ordenes **Inicio > Programas > Rational ClearCase LT Client > Rational ClearCase LT Explorer**.

7.2 Completando el tutorial ClearCase LT

El tutorial inicia al desarrollo de las tareas que se pueden realizar usando ClearCase LT. Antes de iniciar el tutorial, se debe preguntar al gestor o administrador del proyecto o el administrador si se usará UCM.

Para echar a andar el tutorial:

1. Arrancar el explorador de Rational ClearCase LT.
2. En el panel izquierdo, clicar sobre la etiqueta **Toolbox** y después sobre **Getting Started**.
3. Clicar en **Tutorial**.
4. Leer la información introductoria y luego dar un clic sobre el botón apropiado del tutorial.

7.3 Creación de áreas de trabajo y empezando a trabajar

Para encontrar las instrucciones para establecer sus áreas de trabajo:

1. En Rational ClearCase LT Explorer, dar un clic sobre la etiqueta **Toolbox** en el panel izquierdo.
2. Clicar en **Getting Started**.
3. Dar un clic sobre el botón apropiado de desarrollo Software.

Seguir las instrucciones en el panel de información del explorador de Rational ClearCase LT.

Los menús, las barras de herramientas, y las etiquetas en Rational ClearCase LT Explorer contienen los comandos que se usarán para hacer el trabajo. Para aprender más sobre el flujo de tareas en ClearCase LT, se debe utilizar la información del panel de información del Racional ClearCase LT Explorer.

8 Más información sobre ClearCase LT

En este punto se describe la información disponible acerca de ClearCase LT. Se puede obtener más información acerca de la documentación Rational en la página web de Rational Rose <http://www-306.ibm.com/software/rational/support/documentation/>

8.1 Conjunto de información de Rational ClearCase LT

El conjunto de información ClearCase LT incluye la siguiente documentación:

- *Guía de instalación para Rational ClearCase LT.*
- *ClearCase LT Release Notes.*
- *Introducción a Rational ClearCase LT (este manual).*
- *Gestionando proyectos software.*
- *Guía del administrador para Rational ClearCase LT.*
- *Comandos de referencia (Command Referente).*
- *Ayuda de tarea*, disponible en línea.
- *Tutorial*, disponible en línea.

8.2 Informándose Acerca De Tareas Del Desarrollador

En UCM, las tareas típicas de desarrollo incluyen unir un proyecto, trabajando y entregando actividades, y rebasando el área de trabajo. Para la visión general y la información procedural de estas tareas, ver la ayuda en línea.

En la base de ClearCase LT, las tareas típicas de desarrollo incluyen crear y actualizar una vista, registrando la salida y registrando en archivos y directorios, y combinando archivos. Para la visión general y la información procesal para estas tareas, ver la ayuda en línea.

8.3 Informándose sobre las tareas del gestor y el administrador del proyecto

En UCM, los gestores y los administradores de proyecto y de componentes, determinan la política, establecen y promocionan líneas base, y operan la corriente de integración.

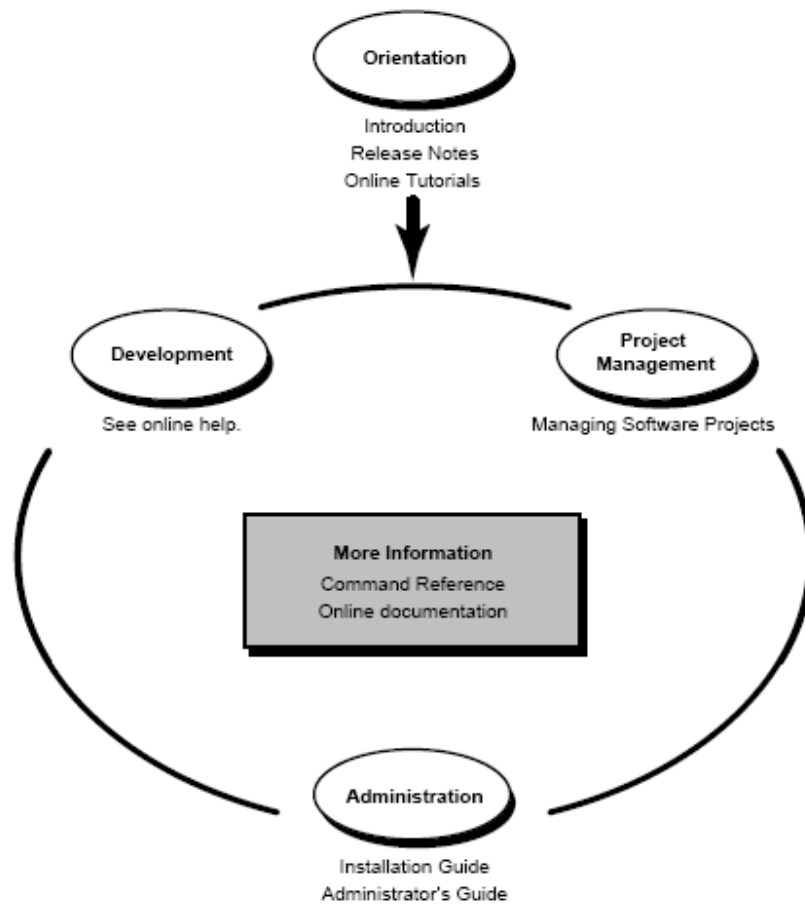
En ClearCase LT, los gestores de proyecto y los administradores determinan la política y crean tipos de la rama, especificaciones de configuración, disparadores, y etiquetas para implementar las políticas.

Para más información, ver la ayuda en línea y *Gestión de Proyectos Software*.

8.4 Informándose Acerca De Tareas Del Administrador

Los administradores crean y mantienen VOBs y manejan el almacenamiento de VOBs y de vistas, licencias, y los registros para ClearCase LT. Para más información acerca de las tareas administrativas, ver la ayuda en línea y *Guía del Administrador para Rational ClearCase LT*.

8.5 Mapa de documentación de ClearCase LT



9 Anexo.

9.1 Convenciones tipográficas

Este manual usa las siguientes buenas costumbres tipográficas:

- **La negrita** sirve para nombres que el usuario puede introducir; Por ejemplo, todos los nombres de comandos, nombres de archivos, y nombres de directorios.
- *La letra cursiva* sirve para variables, títulos del documento, condiciones del glosario, y énfasis.
- Un carácter de imprenta monoespaciado sirve para ejemplos. Dónde la entrada de usuario necesita ser distinguida de salida de programa, **la negrita** sirve para entrada del usuario.
- Los caracteres no impresos aparecen como sigue: <EOF >, <NL>.
- Los nombres cruciales y las combinaciones cruciales son escritas con mayúsculas y aparecen como sigue: SHIFT, CTRL+G.
- [] Los corchetes encierran artículos optativos en las descripciones de formato y de sintaxis.
- { } Las llaves encierran una lista de la cual se debe escoger un artículo en las descripciones de formato y de sintaxis.
- | Una barra vertical separa artículos en una lista de elecciones.

9.2 Documentación en línea

La interfaz gráfica ClearCase incluye un sistema de ayuda en línea.

Hay tres formas básicas para tener acceso al sistema de ayuda en línea: El menú **Help**, el botón **Help**, o la tecla F1. **Help > Contents** proporciona entrada a la colección completa de documentación en línea ClearCase. Para obtener ayuda en un contexto particular, se debe presionar F1. Usando el botón **Help** en las ventanas de diálogo diversas se obtiene información específica de la ventana de diálogo.

ClearCase también provee acceso para las páginas de referencias (las descripciones detalladas de órdenes ClearCase, servicios públicos, y las estructuras de datos