

Introduction

ComponentSoftware RCS (CS-RCS) is a powerful and robust document revision control solution. CS-RCS monitors changes made in files that are accessed by standalone or networked workstations. Based on the widely used GNU RCS, it is fully integrated with Windows 95 and Windows NT. CS-RCS supports multi-user and multi-platform environments, making it the ideal solution for sites that share common files on UNIX and Windows platforms.

ComponentSoftware RCS is based on revision 5.7 of GNU RCS. GNU RCS was developed by the GNU project and is currently used by thousands of users, mainly in the UNIX environment. GNU RCS has an old-fashion command-line user interface, but ComponentSoftware RCS has a modern user-friendly graphical user interface that is fully integrated with the Windows Explorer and shell. This combination of tested power and ease-of-use brings full RCS mastery to your fingertips.

Why Use an RCS?

A Revision Control System (RCS) helps you manage multiple revisions of files by keeping a complete history of the changes performed to the files. This allows you to see how and when a file was changed, or to quickly return to a previous revision of a file. This can be crucial for files that are edited frequently such as programs, word-processor documents, and HTML documents.

How ComponentSoftware RCS Helps You

ComponentSoftware RCS allows you to:

- **Retrieve any revision by any criteria anytime.** Each revision is marked with a unique revision number, revision date, author name, optional symbolic name, and descriptive comment.
- **Conveniently see what was changed between any two revisions.** You can study the evolution of a function throughout its lifetime.
- **Never lose any work because you can always backtrack.** This is critical when there is a deadline and the last revision worked better.
- **Work with binary files.** You can maintain revisions of binary files such as MS-Word documents, pictures, drawings, etc..
- **Save disk space.** Since only the differences between revisions are saved, the archive file size is kept to the minimum size.

In a multi-user environment, ComponentSoftware RCS additionally allows you to:

- **Know who does what and when.** This simplifies communication between users, and smoothes out the whole development process.
- **Avoid two users modifying the same file at the same time.** ComponentSoftware RCS maintains a “lock” for each file.
- **Know when there is an updated revision of a shared document.** ComponentSoftware RCS notifies you whenever a coworker checks-in a new revision.
- **Use variety of servers and networks.** Any file server can be used to keep the RCS archive files. Network connection can be LAN, corporate wide-area network, dial-up connection and the **Internet**.

ComponentSoftware RCS was designed to handle UNIX machines and Windows machines using the same RCS archive repository. In a multi-platform environment, ComponentSoftware RCS additionally allows:

- **Users on UNIX and Windows machines to share common files.** This option increases reusability and stability of products that target both UNIX and Windows platforms.
- **Transparently convert text files UNIX-to-DOS and DOS-to-UNIX.** ComponentSoftware RCS takes care of the line-break sequence that is different for UNIX and Windows/DOS files.

Introduction

This brief tutorial shows you how to perform the basic tasks you need to use ComponentSoftware RCS. We recommend you follow this tutorial through from start to finish. The tutorial takes about 10 minutes.

[Next](#)

Basic Concepts

The ComponentSoftware RCS Repository keeps track of the various revisions of your files. Each file you want to keep track of must be added to the Repository. Whenever a file receives a significant revision, it should be added again to the Repository so that anytime in the future you can go back to that revision of the file. Adding a file again to the repository is called *checking-in* a file.

When you want to edit a file, you should take the latest revision of the file out of the Repository. This way you can make sure you have the file in its correct form. This is called *checking-out* the file.

Next

Set Yourself Up

Before we begin, prepare a sample file:

- Create a new file named TEST.TXT.
- Open Notepad and type a few lines.
- Save the file.

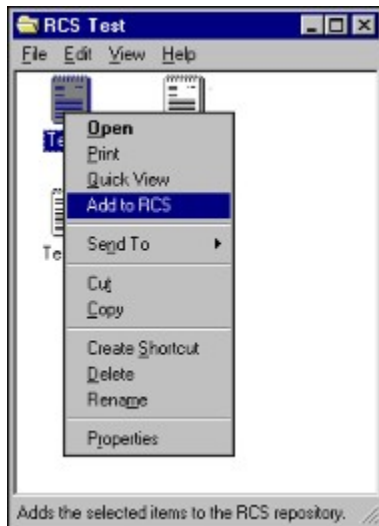
Next

Add a File to the CS-RCS Repository

To have ComponentSoftware RCS keep track of your file, you must add the file to the Repository. Add TEST.TXT to the Repository:

- View TEST.TXT in Windows Explorer.
- Click on the file with the right mouse button to open the context menu, and choose **Add to RCS**.

The current revision of TEST.TXT is added to the Repository.



Next

View the Changes to a File

You can check whether a file has been changed since it was last added to the ComponentSoftware RCS Repository. If the file has been changed, you can see the changes. Check whether TEST.TXT has changed since you last added it to the Repository:

- View TEST.TXT in Windows Explorer.
- Click on the file with the right mouse button to open the context menu, and choose **RCS Status**.



The following message appears:

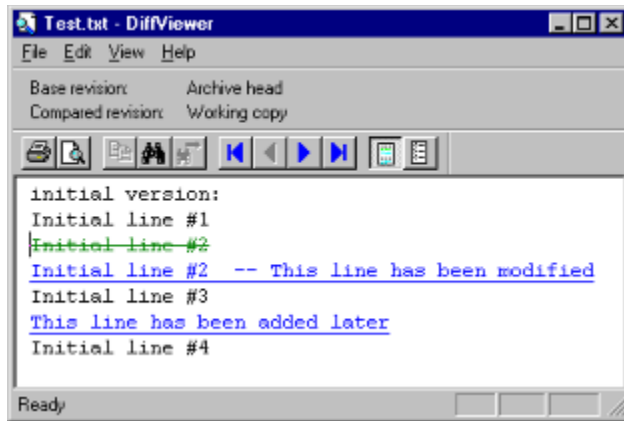


The message indicates that TEST.TXT has not been changed since you last added it to the repository.

Change TEST.TXT and then check again whether TEST.TXT has changed since you last added it to the Repository:

- Using Notepad, add a few lines to TEST.TXT.
- Invoke **RCS Status** again.

A list of the differences between the last version you added to the Repository and the current file appears.



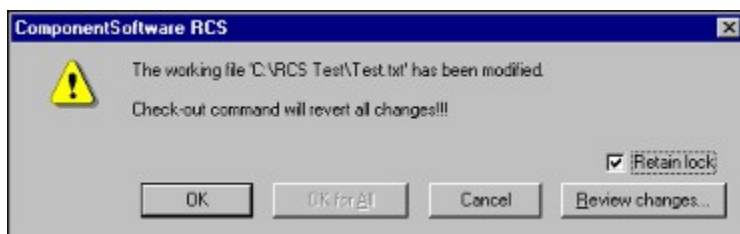
Next

Check-out the Latest Revision of a File

At any time, you can undo all changes you have made since you last added a file to the Repository. This is called *checking-out* a file, and effectively returns the file to how it was when it was last added to the repository. Check-out TEST.TXT:

- View TEST.TXT in Windows Explorer.
- Click on the file with the right mouse button to open the context menu, and choose Check-out.
- Select OK.

TEST.TXT returns to its original revision.



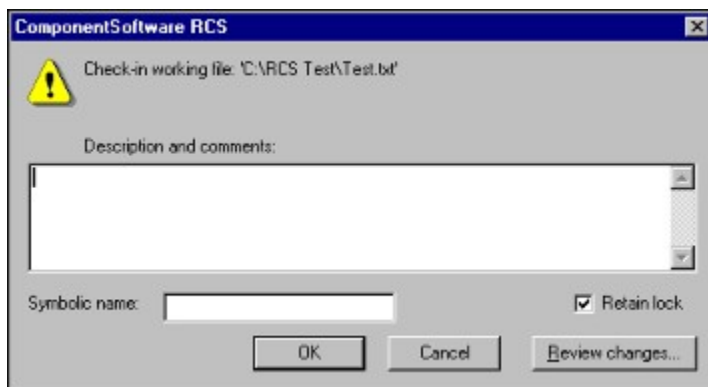
When the Check-out dialog box is open, you can see what was changed since the last revision by pressing **Review Changes**. If you are in a multi-user environment, when you check-in or check-out a file, nobody else can check-in or check-out the same file. If you want other users to be able to check-in or check-out the file, uncheck the **Retain Lock** option.

Next

Check-In a New Revision

As mentioned above, whenever a file has been significantly edited, it should be reentered into the Repository, or *checked-in*. Edit TEST.TXT and check it in:

- Use Notepad to edit TEST.TXT.
- View TEST.TXT in Windows Explorer.
- Click on the file with the right mouse button to open the context menu, and choose Check-in.
- Fill in the Description and comments. Use a symbolic name to emphasize major revisions. Symbolic names cannot include spaces.
- Press OK.



When the Check-in dialog box is open, you can see what was changed since the last revision by pressing **Review Changes**. If you are in a multi-user environment, when you check-in or check-out a file, nobody else can check-in or check-out the same file. If you want other users to be able to check-in or check-out the file, uncheck the **Retain Lock** option.

***Tip:** When checking in a new revision, press Review Changes and copy and paste sections into the Description and Comments field. This will help you keep track of the changes in that version.*

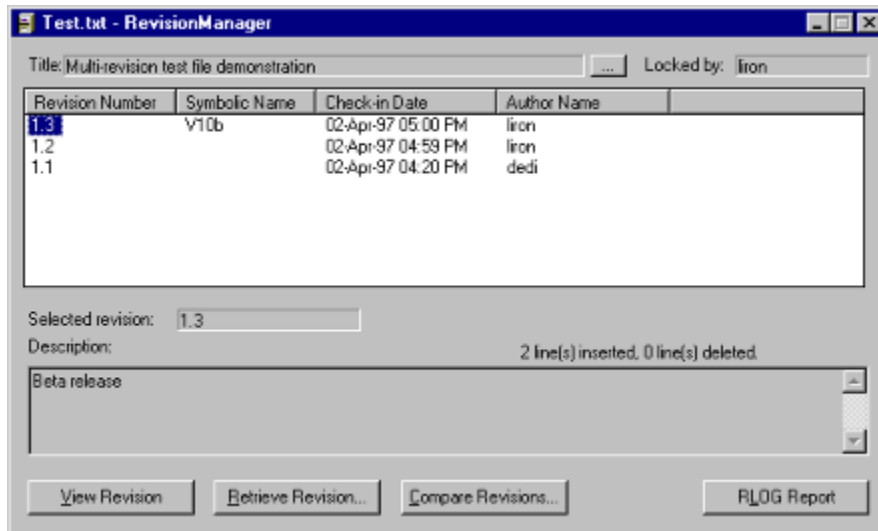
Next

View the Revision History of a File

At any point, you can view a list of the revisions of a file. You can then compare the revisions to one another, and revert to an earlier revision. View the revisions of TEST.TXT:

- View TEST.TXT in Windows Explorer.
- Click on the file with the right mouse button to open the context menu, and choose **Revision History**.

The various revisions of TEST.TXT are displayed. You can sort the list as you wish by clicking on the desired column header. Click again on the same column header to reverse the sort order.



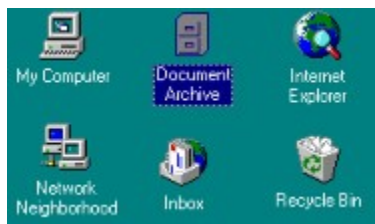
This dialog box gives you many options:

- Select any revision on the list to view the revision's description and comments. Click the right mouse button to edit the revision information.
- Press **View Revision** to view the selected revision.
- Press **Retrieve Revision** to save the selected revision to disk.
- Press **Compare Revisions** to view the changes done between the selected revision to any other revision or the working copy.
- Press **RLOG Report** to view a detailed change report for the file.
- Press ... to set the document title and description.

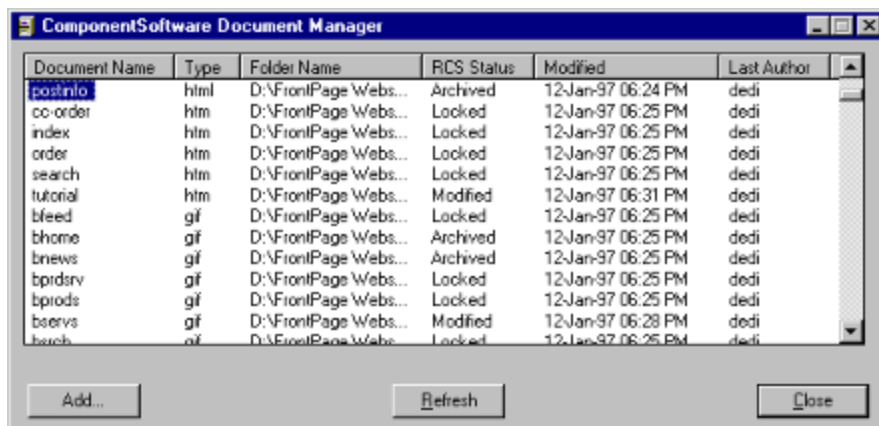
Next

View the Global Picture

ComponentSoftware RCS adds a special icon to your desktop called the *Document Archive*.



At any point, you may activate the *Document Manager* by double-click on the *Document Archive* icon.



You can see the major properties of your document:

- Document name, type and Folder
- Document archive status.
- The last time the document has been checked-in or modified
- The last author of the document.

You can sort the list as you wish by clicking on the desired column header. Click again on the same column header to reverse the sort order. You may click on a document with the right mouse button to open the context menu and apply any RCS command.

The Document Status property gives you the exact state of the document:

- *Locked* – You may change the document. (In a multi-user environment, nobody else may change it.)
- *Modified* – You have changed the document. It is recommended to check-in the document whenever it receives a significant revision.
- *Deleted* – The document exists in the archive but not in your working directory. You may check-out the document in order to get the current revision.

In a multi-user environment, a document may have the following states as well:

- *Archived* – The document is archived. Nobody may change it. (The working copy has a read-only permission.)

- *Blocked By...* – A partner had locked the document. Nobody else may change it till the lock is released.
- *Updated* – The document has been updated in the archive. You should check-out the document in order to get the current revision.

Next

Thank You

Now you have the basic skills necessary to take advantage of the power of ComponentSoftware RCS.
Thank you for participating in this tutorial.

Invoking the Context Menu

The context menu can be invoked in several ways:

- Click on the file with the right mouse button.
- Select the file, and then open the **File** menu.
- If you have a Windows 95 keyboard, select the file and then press the key near the right Ctrl key.

Tip: You can invoke the context menu from the Windows *File Open* dialog box.

Multiple File Operations

Sometimes, it is convenient to perform an operation on several files at once. For instance, if you have edited several files that relate to the same product release, they can all be checked-in at one time. This gives all the files the same check-in information.

To perform a multiple file operation:

- Select all the related files or folders.
- Invoke the context menu and choose the desired option.

Notes:

- Select multiple files and choose the *Retrieve Revision* command to check-out files by specific date or symbolic name.
- Right-click on a working folder and choose *RCS Status* to evaluate the status of files in a working folder tree.

CS-RCS Properties

CS-RCS has been designed for quick-start operation, but many of the CS-RCS properties can be customized. To access the CS-RCS Properties setup program either:

- Right click on the *Document Archive* icon on the desktop and select *Properties*.
- Invoke the Explorer *Start* menu, and select *Programs|CS-RCS|Properties*.

RCS keywords

RCS keywords are special patterns inserted in text files managed by ComponentSoftware RCS. RCS keywords are used to incorporate RCS status information into source files.

For example, the command: `const char *id = "$Id: $"`

Is expanded to:

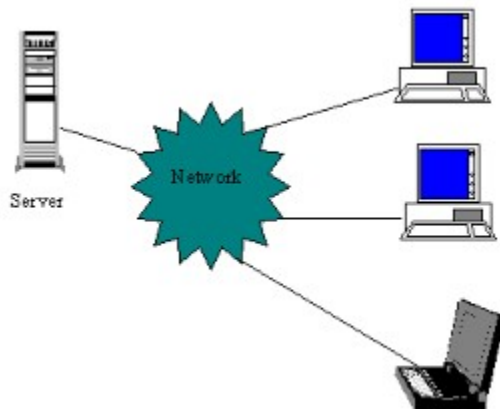
`const char *id = "$Id: key.txt,v 1.2 1997/06/02 12:13:44 Jacob Exp Jane $"`

Click [here](#) for a list of commonly used RCS keywords.

Introduction

Multi-user environment consists of three components:

- The RCS repository server
- The workstations
- The network



The RCS Repository Server

The requirements from the RCS repository server are:

- All workstation must have read and write access to the server's drive.
- The server must support long file names.

Popular servers that meet these requirements are:

- Windows 95 or Windows NT shared (i.e., "exported") folder
- Windows NT Advanced Server
- NetWare Novell (*Note: long file names support should be installed.*)
- SMB server on a UNIX machine (such as public-domain SAMBA)
- NFS server on a UNIX machine.

Notes:

- *You may use the same machine as both server and workstation.*
- *You may refer the shared drive as a normal drive letter (F:, for example) or as an UNC path (\SERVER\RCS-TREE\, for example). The second method is simpler and guaranties unique identification of the shared drive.*

The RCS Workstation

The RCS Workstation

Any machine running Windows 95 or Windows NT can be used as an RCS workstation.

The following example illustrates a typical environment:

The RCS administrator creates and checks-in the following document tree:

```
C:\Project1\...
  C:\Project1\Src
  C:\Project1\Doc
  C:\Project1\FAQ
C:\Project2\...
  C:\Project2\Src
  C:\Project2\WWW
```

From now on, a workstation may retrieve a local copy (called *the working copy*) of the current version. A workstation may retrieve either Project1 or Project2 (or both). A workstation may retrieve a read-only copy or a read and write working copy (called *a locked copy*).

Notes:

- *The working folders must have the same name on all workstations.*
- *If desired, the working folders may reside on a server drive that is private to each workstation. As an example: the working folder may be G:\Project1 while G: is mapped to \\SERVER\User1 on one workstation and mapped to \\SERVER\User2 on another workstation.*

The Network

The Network

You may select the network that best meets your needs:

- **Local Area Network (LAN)** – Teams that are located in the same building should use a LAN to access the RCS repository.
- **Corporate network** – Coworkers that are located on several sites of the same organization should use the corporate network (LAN/WAN) to access the RCS repository.
- **Dial-up connection (RAS)** – Mobile workers can use dial-up lines (regular phone or ISDN) to access the RCS repository server.
- **The Internet** – Coworkers located anywhere in the country or around the globe can use the Internet to access the RCS repository. While the workstations may have local dial-up connection to the Internet, the server must have a permanent connection to the Internet. (Virtual hosting service and connect-on-demand are cost-effective solutions.) The server must be able to export the shared RCS repository tree to the Internet (password protected, of course). This feature is build-in on NT servers and may be easily installed on UNIX servers using a public-domain product named Samba (NetBIOS over TCP/IP). The new Point-to-Point Tunneling Protocol (PPTP), enables Windows 95 and NT clients to communicate securely with an NT Internet server providing a Virtual Private Network (VPN) over the Internet

Tutorial

Ten Easy Steps to Setup a Multi-user Environment

To setup ComponentSoftware RCS in a multi-user environment, follow these steps:

- Activate the setup program and click “Master Setup”. Specify a server drive (such as \\Server\RCS or F:\RCS) for the repository root.
- Follow the tutorial above to get familiar with ComponentSoftware RCS and to verify full functionality.
- Create several additional test files; Add these files to the RCS repository as well.
- Select one or two files and apply "Check-out" with the "Lock" option cleared. That will release the lock from these files. (*Note*: These files are now read-only. You should not modify any working file unless it is locked by you.)
- Activate the “Document Manager” and click “Scan New Files”. Verify the files’ status.
- On another PC, activate the setup program and click “Workstation Setup”. Please verify you use another user name login for this workstation.
- Activate the "Document Manager" and click "Scan New Files". You should see all the archived files marked as *Deleted*. That’s mean there is no local working copy of these file on this workstation.
- Select all files and choose "Check-out". ComponentSoftware RCS will create a local working copy on the workstation PC. File status will be *Blocked by...* for files locked by the administrator, *Locked* for files locked by the workstation user and *Archived* for files without any lock.
- On the workstation, modify and check-in one file.
- Click "Refresh" on the administrator’s "Document Manager" window. The file that has been modified on the workstation should have an *Updated* status. That indicates that your local copy is outdated and you should check-out the updated revision whenever convenient.

Notes:

- Repeat steps 6-8 for as many workstations as desired.
- By default, CS-RCS requires all workstations to maintain the same working directory structure. You may customize this using CS-RCS projects. (Right-click the *Document Manager* icon and choose the *Properties* menu. Then select the *Projects* tab).
- Before checking-in a new revision to the RCS repository, it is recommended to check-out all *Updated* files and to re-test the modifications.

Thank You

Now you have the skills necessary to take advantage of the power of CS-RCS in a multi-user environment. Thank you for participating in this tutorial.

Document Manager Options

- Click on any column header to sort the list as you wish. Click again on the same column header to reverse the sort order.
- Select single or multiple documents. Click the right mouse button to open the context menu and apply any RCS command.
- Press **Add** to add new documents to the RCS repository. (*Note: When using the *local RCS directories* option, select a single file to add the complete RCS directory to the document manager.*)
- Press **Refresh** to refresh the status of all documents. It is convenient to re-sort the list when the refresh operation is completed. (*Note: When working with many RCS files, the refresh operation is sometimes time consuming. Since the status information is cached, further refresh operations will be much faster.*)
- Press **Scan New Files** to scan the RCS repository for new files and refresh the status of all documents.

The Document Status property gives you the exact state of the document:

- *Locked* – You may change the document. (In a multi-user environment, nobody else may change it.)
- *Modified* – You have changed the document. It is recommended to check-in the document whenever it receives a significant revision.
- *Deleted* – The document exists in the archive but not in your working directory. You may check-out the document in order to get the current revision.

In a multi-user environment, a document may have the following states as well:

- *Archived* – The document is archived. Nobody may change it. (The working copy has a read-only permission.)
- *Blocked By...* – A partner had locked the document. Nobody else may change it till the lock is released.
- *Updated* – The document has been updated in the archive. You should check-out the document in order to get the current revision.
- *Unlocked Modified* – Although not locked, this document has been assigned read-write permission and has been modified. Use the check-out command to lock this document or to revert to the current revision. (*Note: Since multiple users may change the same revision, this method should not be used on a regular basis.*)

Revision Manager Options

- Click on any column header to sort the list as you wish. Click again on the same column header to reverse the sort order.
- Select any revision on the list to view the revision's description and comments. Click the right mouse button to edit the revision information.
- Press **View Revision** to view the selected revision.
- Press **Retrieve Revision** to save the selected revision to disk.
- Press **Compare Revisions** to view the changes done between the selected revision to any other revision or the working copy.
- Press **RLOG Report** to view a detailed change report for the file.
- Press ... to set the document title and description.

CS-RCS Properties – About Tab

Use the **About** tab to:

- Verify your CS-RCS version and build number.
- Load the registration key.

CS-RCS Properties – File Types Tab

Use the **File Types** tab to configure how CS-RCS handles any file type:

Text or Binary

CS-RCS does not perform any translation or RCS keyword expansion for binary files.

Viewer (used to view RCS revisions)

You can select the following:

- CS-RCS internal text viewer.
- Program associated with the Windows Explorer *Open* Command.
- User external program.

Revision Difference Analysis Tool

You can select either:

- CS-RCS internal tool – displays rich-text difference analysis and GNU *diff* format for text files. Invokes MS-Word difference tool for *.doc* and *.rtf* documents.
- User external program.

This dialog box gives you the following options:

- Press **Add...** to define a new file type.
- Press **Remove...** to remove the selected file type.
- Press **Edit...** to edit the selected file type.
- Press **Default...** to define the default attributes of new file types.

CS-RCS Properties – Projects Tab

Use the **Projects** tab to define CS-RCS projects. CS-RCS projects can be used in the following scenarios:

- Define a sub-tree of the repository tree as a project. Enables easier navigation and status examination as well as CS-RCS commands execution on a pre-defined set of files.
- Map a project to an alternate working directory. Enables a customized working directory per workstation as well as easily freezing a milestone snapshot on a production server.
- Declare an alternate repository directory as a new project. Enables using CS-RCS to manage several RCS repositories within the corporate network.

This dialog box gives you the following options:

- Press **Add...** to define a new project.
- Press **Remove...** to remove the selected project.
- Press **Edit...** to edit the selected project.
- Press **Workstation...** to define the local workstation working directory of the selected project and to create a desktop shortcut to the project.
- Press **Open...** to open the selected project.

CS-RCS Properties – Users Tab

Use the **Users** tab to define the valid users of CS-RCS.

This dialog box gives you the following options:

- Press **Add User...** to define a new user.
- Press **Remove User...** to remove the selected user.

CS-RCS Properties – Advanced Tab

Use the **Advanced** tab to customize the advanced properties of CS-RCS.

Caution: This form should be used by advanced users with good knowledge of GNU RCS.

This dialog box gives you control on the following properties:

- **RCS suffix** – By default, RCS files are created with a “,v” suffix. (This is the convention used by GNU RCS on UNIX machines.) If you have existing RCS files with another suffix (or with no suffix at all), you may want to change the default CS-RCS suffix.
- **New files' parameters** – Clear this option if you wish to specify the initial revision number and the initial message and description of new CS-RCS files.
- **CS-RCS cache** – CS-RCS caches calculated RCS status for further status evaluation. This option should be always checked unless you have severe local drive space limitations. Click on **Clear Cache** to force re-calculation of all files status. Note: The cache information is usually saved in: C:\Program Files\ComponentSoftware\CS-RCS\Cache
- **Case sensitive user names** – Windows 95 and Windows NT are user-name case insensitive. GNU RCS is user name case sensitive. To resolve this conflict, CS-RCS translate all user names to lower case. If you prefer to work with case-sensitive user names (Windows 95 only) -- enable this option. Note: Before clearing this option, verify all RCS files are *Archived*. Otherwise, you might have problems checking-in or unlocking certain files.

Command-line Interface

Sometimes, it is convenient to perform an operation using command-line interface. For instance, you might wish to invoke ComponentSoftware RCS from development tool such as Visual Basic or Delphi or from batch file. ComponentSoftware RCS supports basic command-line interface and an advanced command-line interface.

Basic Command-line Interface

The basic command-line interface has been designed to enable the integration of ComponentSoftware RCS with development tools such as Visual Basic, Delphi, Visual C++ and programmer's editors.

The basic command-line interface syntax is:

CSRCS command "file-path"

Basic command-line interface commands:

Create – Add a new file to the RCS repository

CheckOut – Check-out a working copy.

CheckIn – Check-in a modified file.

Status – Inspect the RCS status of a file

History – Trace the revisions of a file.

Help – Get on-line help.

Example:

To check-in a modified file to the RCS repository, type in the Explorer **Run...** window:

`CSRCS CheckIn "C:\Project1\Src\test.c"`

Advanced Command-line Interface

The advanced command-line interface has been designed to provide the full power of ComponentSoftware RCS for DOS command windows and batch files. (*Note:* The advanced command-line interface is a superset of the basic command-line interface.)

The command-line interface syntax is:

```
CSRCS command [command-switches] "file1" ["file2" ...]
```

The *file* argument(s) can be either:

- Full path
- Relative path to the current working directory.
- Wildcard argument

Examples:

To get the RCS status of all files in the current directory tree type:

```
CSRCS Status /s *.*
```

To check-in all .c and .h files in the current directory type:

```
CSRCS ci *.c *.h
```

The next sections elaborate the various commands and command options.

Create command

Purpose: Add new files to the RCS repository

Abbreviation: *Add*

Switches:

/s – Handle files in specified directory and all sub-directories.

Exit code: 0 – success; 1 – otherwise.

CheckOut command

Purpose: Check-out working copy of files.

Abbreviation: *co*

Switches:

/s – Handle files in specified directory and all sub-directories.

/l – Lock the files.

/u – Unlock the files.

/q – quiet mode. (Do not display dialog box and status messages)

Exit code: 0 – success; 1 – otherwise.

CheckIn command

Purpose: Check-in modified files. Assign symbolic name to revision.

Abbreviation: *ci*

Switches:

/s – Handle files in specified directory and all sub-directories.

/l – Lock the files.

/u – Unlock the files.

/m "message" – Use *message* to describe the checked-in revision.

/nname – Use *name* as the checked-in revision symbolic name.

/q – quiet mode. (Do not display dialog box and status messages)

Exit code: 0 – success; 1 – otherwise.

Status command

Purpose: Display the files' RCS status

Abbreviation: *st*

Switches:

/s – Handle files in specified directory and all sub-directories.

Exit code: 0 – success; 1 – otherwise.

History command

Purpose: Trace the revisions of a file.

Abbreviation: *his*

Switches:

None

Exit code: 0 – success; 1 – otherwise.

Note: The history command is valid for single file only.

Retrieve command

Purpose: Retrieve archived revision of files.

Abbreviation: *ret*

Switches:

/s – Handle files in specified directory and all sub-directories.

/d "date" – Retrieve the last revision created before a specific date. (if any)

/nname – Retrieve a revision marked with a symbolic name. (if any)

/q – quiet mode. (Do not display dialog box and status messages)

Exit code: 0 – success; 1 – otherwise.

Notes:

- */d* and */n* are mutually exclusive. If none of them is specified, the head revision is retrieved.
- Valid date formats are: DD-MMM, DD-MMM-YY, DD-MMM-YY HH:MM or HH:MM. For example: 15-Jun-97 10:45 ; 16-Jun ; 20:00 (today)

Help command

Purpose: Get on-line help.

ComponentSoftware RCS Frequently Asked Questions (FAQ)

This FAQ has been updated on July 15, 1997. For the up-to-date FAQ, visit our web site at:
<http://www.ComponentSoftware.com/csrfaq/>

ComponentSoftware RCS and GNU RCS FAQ

What are the benefits of CS-RCS being based on GNU RCS?

CS-RCS is based on GNU RCS. CS-RCS archive files are 100% GNU RCS compatible. This architecture grants several benefits:

- Stability – GNU RCS is robust revision control system used by many thousands of developers.
- Interoperability – GNU RCS is available for many platforms and systems.
- Freedom – Since GNU RCS is de-facto standard for revision control, most configuration management products can import GNU RCS archive files. As a result, you have the freedom to switch to any product whenever you decide your needs have changed and CS-RCS does not fulfill them any more.

Can I use CS-RCS in conjunction with GNU RCS?

Since CS-RCS archive files are 100% compatible with GNU RCS, GNU RCS can be used in conjunction with CS-RCS. You may download the command-line port of GNU RCS to Windows 95 and NT from: <http://www.coast.net/cgi-bin/coast/dwn?nt/pgmtools/rcs57pc1.zip>

Note: CS-RCS is based on GNU RCS version 5.7.

Is CS-RCS GPL complaint?

The CS-RCS front-end and GNU RCS are two separate applications that run as separate Windows processes. The CS-RCS application is proprietary, but the GNU RCS application is distributed under the terms of the GNU GENERAL PUBLIC LICENSE (GPL). CS-RCS is not linked with RCS because such linking would violate the terms of the GPL. The Free Software Foundation affirms that running the two as separate processes, with a clean, well-defined general interface between them, justifies the view that they are two separate programs.

ComponentSoftware RCS Evaluation and Registration FAQ

Are there any differences between the evaluation version and the registered version?

No. CS-RCS is marketed in a true try-before-you-buy method. With this method you don't have to trust brochures and biased articles or to evaluate a limited demo; you evaluate the actual product and verify it meets your needs. When you decide to purchase the product, you are emailed a registration key. CS-RCS has been designed for simple installation and evaluation. Single user installation and following the tutorial is just 15 minutes. Multi-user installation and following the tutorial is about 30 minutes.

What happens when the 30 days evaluation period expires?

When the 30 days trail period expires, CS-RCS is blocked and you have to uninstall it. Your archive files are not affected and may be used by any port of GNU RCS. You may continue using CSDiff (the file comparison tool included with CS-RCS) free of charge.

Note: If you need an extended evaluation period -- please tell us.

Do I commit anything by evaluating CS-RCS or by using technical support?

You commit nothing. If you find that CS-RCS does not fulfill your needs, just uninstall it.

What is your technical support policy?

We provide comprehensive technical support for both registered users and evaluation sites. We believe that technical support is an important factor for product purchase and offer a try-before-you-buy method for the technical support aspect as well.

What is your upgrade policy?

When upgrading, you get a full credit for your existing license value (excluding shipping & handling fees).

ComponentSoftware RCS Installation FAQ

Does setup and/or uninstall affect the archive files?

CS-RCS archive files are 100% GNU RCS compatible. Therefore, the archive files do not carry any CS-RCS specific information and are not affected by the setup or the uninstall process.

How do I use CS-RCS over the Internet?

You may locate your repository files on an Internet or Intranet server and access the repository in a method called "NetBIOS over TCP/IP".

The setup of client Windows machines to use "NetBIOS over TCP/IP" is quite simple. (Just define server name and IP address in the LMhosts file. On NT 4.0 verify that the "Enable LMhosts Lookup" option of the "TCP/IP Properties" is checked.)

The server must run SMB server protocol. This feature is build-in on NT servers and can be easily installed on UNIX servers using a public-domain product named Samba. (Samba home page URL is: <http://samba.canberra.edu.au/pub/samba/>)

The new Point-to-Point Tunneling Protocol (PPTP), enables Windows 95 and NT clients to communicate securely with an NT Internet server providing a Virtual Private Network (VPN) over the Internet

What is the structure of the central repository tree used to store the archive files?

CS-RCS stores archive files in the following repository tree structure:
<repository root>\<drive>\<directory>

As an example, if the repository root is F:\RCS then:

```
c:\project1\src -> f:\rcs\c\project1\src
c:\project1\inc -> f:\rcs\c\project1\inc
c:\project2\src -> f:\rcs\c\project2\src
\\server1\doc\project1 -> f:\rcs\server1\doc\project1
```

Tip: Add a sample working file to the RCS repository. Using the Windows Explorer, discover the repository tree structure. Later, you may delete the sample files.

How do I import existing GNU RCS files to CS-RCS?

- Create the desired directory structure (see above).
- Copy your files to the desired directories.
- Press Scan New Files in the Document Manager window. The newly added files should be marked as Deleted, means there is no local working copy for these file. Select these files and apply Check-out.

Note: When importing files from UNIX systems by FTP, make sure to use binary transfer mode.

Otherwise, you get a spurious "^M" on your text files. For simultaneous use of the archive file on Windows and UNIX platforms, see below.

Why does the Scan New Files command take so long?

The Scan New Files command searches the complete repository tree for new RCS files and then computes their CS-RCS status. This operation is sometimes time consuming. Since this information is cached, further Scan New Files and Refresh commands will do much faster.

How do I change the repository root? How do I switch from local RCS directory to central repository?

- Reinstall CS-RCS and specify your desired repository root.
- Create the desired directory structure (see above).
- Copy your old files to the desired directories.
- Press Scan New Files in the Document Manager window. The newly added files should be marked as Deleted, means there is no local working copy for these file. Select these files and apply Check-out.

How do I import MKS RCS files to CS-RCS?

Since it depends on the MKS RCS version used to create the files, please contact our technical support staff. Please attach a sample archive file.

Can I install CS-RCS on NT 3.51?

CS-RCS is fully functional under NT 3.51, except for the Explorer and Shell integration,

How do I install long file names support on Novell servers?

From Novell FAQ:

Q. How can I use long filenames for Windows 95 or NT Workstation with NetWare 4.1?

A. At the server console :

volumes

Load OS2.NAM

ADD NAME SPACE OS2 TO VOLUME volumename

volumes

Make sure you are using the Client 32 software, which supports long filenames.

How do I uninstall CS-RCS?

- Select the Add/Remove Programs icon on the Control Panel folder.
- Select CS-RCS and click Add/Remove...

How do I remove the CS-RCS Explorer context menus?

- Activate *RegEdit* (Start | Run... and type regedit)
- Remove the key: HKEY_CLASSES_ROOT*\shellex\ContextMenuHandlers\CS-RCS
- Exit *RegEdit*

How do I remove the Document Archive icon from the desktop?

- Activate *RegEdit*. (Start | Run... and type regedit)
- Remove the key: HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Explorer\Desktop\NameSpace\{a6ccc050-e0bd-11cf-a333-60a052000000}
- Exit *RegEdit*
- Press F5 on the desktop (or simply reboot).

Windows/UNIX Installation FAQ

What do I have to change in the Samba configuration in order to use CS-RCS?

You must define the delete readonly flag value to yes. Otherwise, multiple users won't be able to check-in/check-out files.

To work with a full case sensitive file-system, define the case sensitive and preserve case flags to yes.

Below please find a sample Samba configuration file (smb.conf):

```
[homes]
comment = Home Directories
browseable = no
read only = no
create mode = 0750
case sensitive = yes
preserve case = yes
delete readonly = yes
```

How do I map active UNIX RCS files to CS-RCS repository?

This mapping is done using UNIX "ln" command.

For example:

```
Working directory – c:\project\src
UNIX RCS files directory – /archive/project1/src
CS-RCS repository root: /users/CS-RCS
```

Type on the UNIX terminal:

```
ln -s /archive/project1/src /users/CS-RCS/c/project/src
```

How do I migrate from SCCS to CS-RCS?

You may convert your SCCS files to RCS using a script named sccstores (available in few software sites on the net). Then you can use either the NT port of GNU RCS or CS-RCS.

ComponentSoftware RCS How To... FAQ

How can I invoke CS-RCS from other applications or batch files?

CS-RCS supports three levels of command-line interface:

Basic interface:

This is a very simple command line interface designed mainly to enable the integration of CS-RCS with development tools such as Visual C++, Visual Basic and Borland Delphi. Click [here](#) for full details and syntax.

Advanced interface:

The advanced command-line interface has been designed to provide the full power of ComponentSoftware RCS for DOS command windows and batch files. Click [here](#) for full details and syntax.

Raw-level interface:

This interface should be used by programmers with good knowledge of GNU RCS. With this interface you have the full power to manipulate the archive files. Since CS-RCS archive files are 100% compatible with GNU RCS, the command-line port of GNU RCS is used in conjunction with CS-RCS.

To download the command-line port of GNU RCS to NT visit: <http://www.coast.net/cgi-bin/coast/dwn?nt/pgmtools/rcs57pc1.zip>

How do I retrieve a baseline of my documents?

Select the desired documents and apply the Retrieve Revision command. You may retrieve by the following criteria:

- The current ("head") revision.
- A "snapshot" of the documents as of particular time and date.
- By symbolic name as set by the check-in command.

Note: It is recommended to assign symbolic names to emphasize major revisions. To do so, select all your Modified and Locked files and apply the Check-in command.

What are RCS keywords? How do I use them?

RCS keywords are special patterns inserted in text files managed by CS-RCS. RCS keywords are used to incorporate RCS status information into source files.

For example, the command: `const char *id = "$Id: $"`

Is expanded to:

`const char *id = "$Id: key.txt,v 1.2 1997/06/02 12:13:44 Jacob Exp Jane $"`

Click [here](#) for a list of commonly used RCS keywords.

Note: Since keyword expansion can slow-down check-in and check-out operations and cause unnecessary compilations, you may want to use keywords on specific files only.

Common RCS Keywords

\$Header: \$ -- A standard header containing the full pathname of the RCS file, the revision number, the date and time, the author, the state, and the locker (if locked).

\$Id: \$ -- A standard header containing the RCS file name, the revision number, the date and time, the author, the state, and the locker (if locked).

\$Log: \$ -- Complete history of the file including the log message supplied during checkin.

\$Author: \$ -- The login name of the user who checked in the revision.

\$Date: \$ -- The date and time the revision was checked in.

\$Locker: \$ -- The login name of the user who locked the revision (empty if not locked).

\$Revision: \$ -- The revision number assigned to the revision.

\$Name: \$ -- The symbolic name used to check out the revision, if any.

\$RCSfile: \$ -- The name of the RCS file without a path.

\$Source: \$ -- The full pathname of the RCS file.

Note: Since keyword expansion can slow-down check-in and check-out operations and cause unnecessary compilations, you may want to use keywords on specific files only.

