

**Team
Coherence**

www.teamcoherence.com

Command-line Reference

© 1995-2015 MCN Software Ltd

TC Command-line Help

© 1995-2015 MCN Software Ltd

All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of the publisher.

Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

Printed: May 2015 in (wherever you are located)

Publisher

*Quality Software Components
6 Suttie Way
Bridge of Allan
Stirlingshire
FK9 4NQ
Scotland*

*E-Mail: info@qsc.co.uk
Website: <http://www.qsc.co.uk>
Fax: +44 (0)1786 834908*

Table of Contents

Foreword	0
Part I Introduction	4
1 What is Team Coherence?	5
2 How to buy Team Coherence	6
Part II Commands	8
1 Command line Options	8
-? Command-line Option	8
@ Command-line Option	8
-A Command-line Option	8
-B Command-line Option	8
-C Command-line Option	9
-D Command-line Option	9
-E Command-line Option	10
-F Command-line Option	10
-G Command-line Option	11
-I Command-line Option	11
-K Command-line Option	11
-LC Command-line Option	12
-M Command-line Option	12
-N Command-line Parameters	12
-O Command-line Option	12
-P Command-line Option	13
-R Command-line Option	13
-S Command-line Option	13
-T Command-line Option	14
-U Command-line Option	15
-V Command-line Option	15
-W Command-line Option	16
-X Command-line Option	16
-Y Command-line Option	16
2 Commands	17
About Team Coherence	17
Add	18
AddConnection	19
AttachLabel	19
CD	20
CheckIn	21
CheckOut	22
Connect	22
ConnectInfo	23
Create	23
CreateLabel	24
CreateView	25
Delete	25
DeleteConnection	26
DeleteLabel	26
DeleteView	27
DetachLabel	27
Directory	28
Get	29

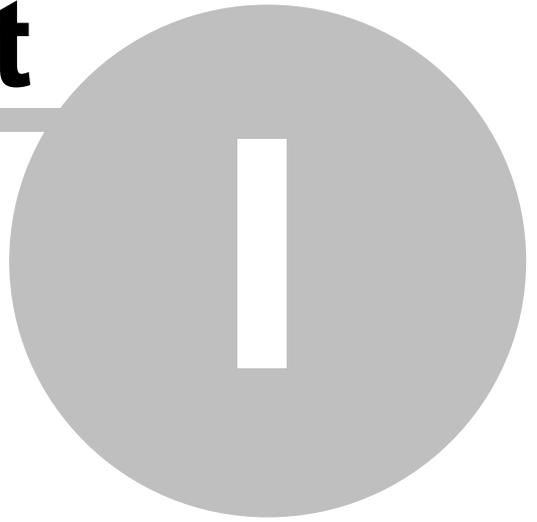
Help	29
HistoryReport	30
ListConnections	31
ListLockedFiles	31
Promote	32
Rename	33
RenameLabel	33
SetView	34
Sync	34
Undo CheckOut	35
UpdateConnection	36
UpdateView	36
Views	37
Whoami	37
WorkFolder	38

Index

39

Introduction

Part



1 Introduction

This help file introduces and describes the Team Coherence Command-line utility: **TC.exe**. The command line tool allows you to execute several Version Control and maintenance tasks using batch files and macros.

Carrying Out Commands on the Command Line

You can start a command shell from Windows 95 or Windows NT and type Team Coherence commands from there. To do this, the Team Coherence Client bin folder must be on your path (set with the PATH environment variable).

You can also use the **Run** command on the Windows **Start** menu to run a specific command. For example, you can type **tc Get *.*** to get all files in the current project/folder.

Command-Line Syntax

Command-line commands have the following general syntax:

```
tc command [one or more items] [one or more options]
```

What an item is depends on the command. For instance, for the **Get** command, an item is a Team Coherence file or project; for the **Add** command, an item is an operating system file.

In almost all cases where you can specify an item to act on, you can specify more than one item. For instance, any of the following are valid ways of executing the **Get** command:

```
tc Get HELLO.PAS
tc Get Test.pas TestProj.dpr APackage.dpk
tc Get *.pas *.dpr
tc Get *.*
```

The command line is case-insensitive, that is, the command *tc Get HELLO.PAS* can also be typed as *tc get Hello.pas*.

Linux

Under Linux, because of a limitation of the installer, the commandline call is set to **tcx**. If you want, the script can be renamed to **tc** to match this help file.

Current Project/Folder

The current Project/Folder is maintained between TC sessions. Therefore, the next time you start the command line utility the position in the hierarchy is the same as the last time you executed the command.

File Groups

When listing files in a project or folder, you may not see all files that are archived. If file groups are enabled in the Team Coherence repository you are connected to, only the main file of each grouping will be displayed. Similarly, when adding files to a folder, only the main file of each grouping will be displayed. The associated files are automatically added.

For more information on File Groups, see the main Team Coherence help file: [TCVcs.hlp](#)

1.1 What is Team Coherence?

Team Coherence is the simplest to use Software Configuration Management (SCM) solution available today. Rather than turning SCM into a black art that requires weeks of training, enforces unacceptable restrictions on developers, and generally interferes with the development and testing processes, we have designed Team Coherence to be as easy to use as possible.

We have spent many years, and worked with many development teams, to make Team Coherence a real-world solution to what can be a complex process. Regardless of the size of your organization, Team Coherence helps you organize, manage, and protect your software development projects on every level - from storing and tracking changes to individual files, to managing and monitoring an entire development cycle.

Team Coherence has been designed with the future in mind. It is easily extendable through addins and has a modular architecture to allow for future enhancement.

1.2 How to buy Team Coherence

You can order Team Coherence online directly from our home page.

Home page

www.teamcoherence.com

E-Mail support

support@teamcoherence.com

Post

MCN Software Ltd
6 Suttie Way
Bridge of Allan
FK9 4NQ
Scotland

Fax

+44 (0)1786 834908

Commands

Part



2 Commands

2.1 Command line Options

Command-line options alter the way a particular command operates, and can be used to pass additional parameters to a command. Certain options can only be used with certain commands and a list of allowable options is shown with each command.

If an option is used with a command that doesn't support it, that option will be ignored.

2.1.1 -? Command-line Option

Use the `-?` option with any command to get Help for that command. You can also use the `-H` option to obtain online Help.

For example, to display help for the `Checkout` command, type the following:

```
tc checkout -?
```

2.1.2 @ Command-line Option

Use the `@<filename>` option to specify a file containing the list of files or archives to operate on. This option must be followed by the name of an existing file containing either a list of local file names, or a list of valid Team Coherence archive names.

For example, to checkout the files listed in `c:\filelist.txt`, to their default working folder:

```
tc checkout @c:\FileList.txt -GW
```

Note: the file must contain a list of fully qualified filenames. Wildcards will not be expanded.

2.1.3 -A Command-line Option

Use the `-AD` option for any command that requires or allows a date range.

The following table lists ways to use the `-C` option:

Option	Description
<code>-ADF<date></code>	From Date
<code>-ADT<date></code>	To Date

The [HistoryReport](#) command allows you to specify a date range to filter the report to revisions valid between two dates. Use the `-AD` option to specify these dates.

`<date>` should be enclosed in quotes if spaces are present.

2.1.4 -B Command-line Option

Use the `-B` option with the [CreateView](#) and [UpdateView](#) commands make the View you are creating or updating Shared (accessible by all users).

For example, the following command will create a sharable View called `TestView` based on two projects (`Project1` and `Project2`), and using Version Label `1.2.3.4` as the basis:

```
tc CreateView TestView "Description of TestView" Project1 Project2 -VL"1.2.3.4" -B
```

2.1.5 -C Command-line Option

Use the **-C** option for any command that requires a comment. If you do not specify the **-C** option, TC requests a comment for each item specified on the command line. If you do specify the option, all items receive the same comment.

The following table lists ways to use the **-C** option:

Option	Description
-C	All items receive the same comment
-C-	No comment is used
-Ctext	Uses the string text as the comment for all items
-C@File.txt	Uses the contents of File.txt as the comment for all items

A few operations, for example **CheckOut**, **CheckIn**, and **CreateLbl** will prompt for a comment by default. If running these commands in a batch file you should use an option other than the **-C** option so that user input is not required.

If you want to enter a carriage return when entering comments at the command line, use: `\n`

Defaults

TC gives you the option of providing a comment for each file you check out or in. This is the default behavior. The comment tells other users why you checked the file out or what changes were made when checking in

You can modify the default behaviour of TC by specifying a default setting for comments in the TC.INI file. Add the **Comment=** line to the [Defaults] section of the TC.INI file specifying one of the following:

Does not prompt for a comment:

```
Comment=-C-
```

Prompts for one comment for all files:

```
Comment=-C
```

Reads the specified text file as a comment:

```
Comment=-C@filename
```

Uses the specified comment:

```
Comment=-Ccomment
```

To override the default, specify another **-C** option on the command line.

2.1.6 -D Command-line Option

Use the **-D** option when defining or updating a connection to a repository. The **-D** option allows you to specify the Dial Up Networking (DUN) parameters required if the connection has to automatically dial a predefined connection.

The following table lists ways to use the -D option:

Option	Description
-DCname	Specifies that the connection uses the DUN connection defined by name
-DUusername	If the DUN connection requires a username , use this parameter.
-DPpassword	If the DUN connection requires a password , use this parameter.

When updating or creating connections to a specific repository, these options allow you to define any DUN connection parameters required. Note that these settings will only apply to the Windows version of the command-line tool.

2.1.7 -E Command-line Option

Use the **-E** option with the Get command to force the local file to be made writable after a successful Get action.

For example, the following command will perform a Get on the file `info.txt` to its default working folder, and will make the workfile writable:

```
tc get info.txt -GW -E
```

2.1.8 -F Command-line Option

When adding or checking in files, TC needs to know what to do when it encounters files that have not changes and needs to know what to do with the local file if the file is successfully checked in. This is handled using the -F options.

The following table lists ways to use the -F option.

Option	Description
-FU	Causes unchanged files to be checked in.
-FU-	Unchanged files will not be checked in.
-FL	Once checked in, the archive remains locked so that the local file can be modified.
-FM	If assigning a Version Label, causes the existing label to be moved to the newly created revision.
-FB	Forces a Branch revision to be created.
-FR	Causes the local file to be made read-only after successful check in
-FW	Causes the local file to be left writable after checkin. Not recommended as a lock is not maintained.
-FD	Deletes the local file after a successful checkin.

By default, if none of these options are specified, the command will use -FU and -FR

2.1.9 -G Command-line Option

The **Get** command copies files from a Team Coherence archive into a folder. TC gets files in response to the **Get** command and also in response to other commands such as **CheckOut**.

For any of these commands, the **-G** option is used to change the nature of the Get operation.

The following table lists ways to use the **-G** option.

Option	Description
-GW	This option will cause the files to be extracted to their default working folder.
-GLpath	Copies a file to the specified folder, not to the current or working folder

By default, if neither of these options are specified, the files will be copied to the current folder.

The default behaviour can be changed to always default to the Working Folder (**-GW**) for the file by enabling the keyword **ForceWorking** in the TC.INI file. To enable it as the default, add the line `ForceWorking=1` to the [Defaults] section of TC.INI. To disable it as a default, set the value to 0. For example, the following enables **-GW** as the default:

```
[Defaults]
ForceWorking=1
```

If set, you can disable `ForceWorking` for a specific command using **-GW-**, or using the **-GLpath** option instead.

2.1.10 -I Command-line Option

Use the **-I** option with the [HistoryReport](#) command to specify which revision columns to include in the report.

Valid values are a combination of the following constants:

```
hrRevisionName = 1;
hrComments = 2;
hrTimestamp = 4;
hrAuthor = 8;
hrLockedBy = 16;
hrVersionLabels = 32;
hrPromotionLevels = 64;
```

For example, to include the Revision Name, Comments, Timestamp and Version Labels, use the following value:

```
tc HistoryReport *.pas -R -AO"c:\report.htm" -I39 -VF"1.2.3" -VT"1.3.4"
```

2.1.11 -K Command-line Option

When adding or updating a connection ([AddConnection](#), [UpdateConnection](#)), the **-K** option allows you to specify the unique key required to access that connection.

The following table lists ways to use the **-K** option.

Option	Description
-Kkey	The key will be used when accessing this connection

2.1.12 -LC Command-line Option

Use the **-LC** option with the `CheckIn` command to force the use of checkout comments during checkin.

For example, the following command will checkin `info.txt`. If comments were attached during the checkout of this file they will override the comments passed in the command:

```
tc checkin info.txt -C"Checkin comments" -LC
```

2.1.13 -M Command-line Option

When assigning Version Labels using the **AttachLbl**, **Get** or **Checkout** commands, you can use the **-M** option to move an existing instance of that Version Label, if any, to the new revision.

For example:

```
TC AttachLbl "Existing Label" -M
```

will cause any existing label with the name **Existing Label** to be removed from a file before attaching it to the tip revision of the file.

2.1.14 -N Command-line Parameters

Use the **-N** option when updating a connection to a repository. The **-N** option allows you to specify the Description, Host, and Port of the named connection..

The following table lists ways to use the **-N** option:

Option	Description
-NDdescription	Changes the description of the specified connection
-NHhost	Changes the host for the specified connection. Host can be either a hostname or an IP address
-NPport	Changes the port number used in the specified connection.

2.1.15 -O Command-line Option

The **-O** option controls the output from commands that might display large amounts of information. If you do not specify **-O**, the information scrolls by on the screen too quickly to be read.

The following table lists ways to use the **-O** option.

Option	Description
-O	All output is displayed
-O-	Only errors are displayed
-O@file.txt	Errors are displayed and all output is appended to the specified file
"-O&file.txt"	No output is displayed and all output is appended to the specified file. Note that the quotation marks are required on Windows NT platforms, and optional on others. If the named file already exists, output is appended to that file.
-O&-	No output is displayed

2.1.16 -P Command-line Option

When adding a new Version Label ([CreateLabel](#)), the **-P** option allows you to associate the newly created Version Label with a root Project.

The following table lists ways to use the **-P** option.

Option	Description
-PProjectName	The newly created Label will be associated with ProjectName

2.1.17 -R Command-line Option

The **-R** option is used with TC commands that operate on projects and folders; it makes the commands recursive to subfolders.

Normally, commands act only on the specified folder and the files in it. For example, if you get a folder, you get all the files that are in that folder. However, subfolders and the files and subfolders inside them are ignored. When you use the **-R** option to make a command recursive, the command acts on the entire folder hierarchy.

For example, if you type `TC Get //MyProject/Root -R`, folders are created for that folder's subfolders, and the subfolders inside of them, and so on.

2.1.18 -S Command-line Option

The **-S** option enables **SmartMode**. This is used by the Create command to force selection of the newly created folder or project. For example:

```
TC Create NewFolder -S
```

will create a new subfolder called NewFolder and select this as the current folder.

SmartMode can be set as the default for commands that support it. To enable SmartMode as the default add `SmartMode=1` to the [Defaults] section of the TC.INI file. To disable it as a default, set the value to 0:

```
[Defaults]
SmartMode=1
```

To disable SmartMode for a specific command, use **-S-**

2.1.19 -T Command-line Option

The -T option causes the command to terminate as soon as an error at a specific level occurs. If this option is specified on the command-line, and an error occurs, the command will return a non-zero error code:

```
TC CheckOut *.pas *.dfm -T
```

will attempt to checkout all files with a .pas or .dfm extension. If any error occurs, the call terminates and returns a non-zero value. If it succeeds, the call returns zero.

```
TC Get *.pas *.dfm -VA"1.2.3.4" -T2
```

will attempt to checkout all files with a .pas or .dfm extension, and assign the Version Label to all checked out files. If any error occurs at level 2 or above, the call terminates and returns a non-zero value. If it succeeds, the call returns zero.

If you do not specify -T on the commandline, errors are output to the screen, but the command continues execution for the rest of the items and will return 0.

The following table lists ways to use the -V option.

Option	Description
-T	Terminates the command if any error occurs, regardless of the significance (same as -T1)
-T1	Same as -T
-T2	Terminates the command if a level 2 or above error occurs.
-T3	Terminates the command if a level 3 error occurs.

When a command is issued it is sometimes works on a large number of files and inevitably non-significant errors may be raised. For example, checking out a Project and requesting a specific Version Label can cause an error to be raised (level 2) for the files that do not contain that label. Also, checking a file in that hasn't changed can raise the (level 1) error 'File has not changed'. In some cases, these errors are irrelevant to what the user is doing. In this case, you can specify that the command will terminate only when an error at a specific level or above occurs using the switch above.

The following table lists the error levels:

Level	Description
1	Minor error: Err_FileNotChanged, Err_RevisionBranched, Err_FilesTheSame
2	Intermediate error: Err_VersionAlreadyAssigned, Err_FileNotLockedBy, Err_VersionNotFound, Err_FileNotLocked, Err_FileLockedByUser, Err_FileLocked, Err_HasOfflineLock, Err_FileFrozen, Err_ActionCancelled, Err_Cancelled, Err_FilesWritable, Err_NotCheckedOutFromProject, Err_CannotFindRevision, Err_VersionDoesNotExist, Err_CannotFindVersion
3	Major error: all other error codes

The full list of error codes can be found by downloading the Team Coherence API.

2.1.20 -U Command-line Option

When defining or updating connection settings, using the **-U** option forces all data between the client and server to be encoded using UUEncode. This is not normally necessary when the client and server are on the same internal network, but can be required when accessing a remote repository.

If you are having issues with accessing a remote repository, try applying this setting and see if it helps.

To disable UUEncoding, use the **-U-** option.

See Also:

[AddConnection](#)
[UpdateConnection](#)

2.1.21 -V Command-line Option

The **-V** option allows you to select earlier versions of a file and also can be used to assign Version Labels during certain commands. The **-V** option can be used to select either a specific revision or a revision/file containing a Version Label.

The following table lists ways to use the **-V** option.

Option	Description
-VR1.2	Selects revision 1.2
-VL"My Label"	Selects the revision/file with label My Label attached. If used in the context of CreateView and UpdateView , causes the basis of the View to be My Label .
-VA"My Label"	When a command allows you to assign a label, for example the CheckIn command, this option allows you to specify the version label to assign.
-VCA"New Label"	Similar to the -VA option, but creates a new version label.
-VD"DateTime"	AttachLabel only. Allows you to specify the revision by a timestamp rather than a Version Label. The revision selected will be the latest revision created before or on the DateTime specified. DateTime should be enclosed in double-quotes in your local format. e.g. - VD"29/12/2003 23:00"
-VP"Promotion Level"	CreateView and UpdateView only. If basing the View on a Promotion Level, use this command switch.

Other uses:

The history report uses the following flags:

Option	Description
-VF"1.2.3"	Lists all revisions starting at the one containing the version label 1.2.3
-VT"1.3.4"	Lists revisions up to the revision containing version label 1.3.4 . Should be combined with -VF to specify a range.
-VI	If included, the revision containing the 'from' version label will not be output in the report.

2.1.22 -W Command-line Option

By default, when the **Get** or **CheckOut** commands encounter a non-readable local file, the action for that command fails for safety reasons.

If you want TC to overwrite any writable files it encounters, specify the **-W** option.

2.1.23 -X Command-line Option

Use the **-X** option when defining or updating a connection to a repository. The **-X** option allows you to specify the SOCKS parameters required, if any, if the connection has to connect through a SOCKS compliant proxy.

The following table lists ways to use the **-X** option:

Option	Description
-XVversion	Specifies the version of SOCKS that should be used. Can be one of V4, V4A, V5
-XHhostname	The hostname of the SOCKS server.
-XPport	The port that the SOCKS server uses.
-XUusername	The username required to access the SOCKS server.
-XWpassword	The password required to access the SOCKS server.

When updating or creating connections to a specific repository, these options allow you to define any SOCKS connection parameters required.

2.1.24 -Y Command-line Option

The **-Y** option specifies a username in case you want to execute a command as a user other than yourself.

The following table lists ways to use the **-Y** option.

Option	Description
-YLindsey	Specifies a username
-YLindsey,MyPass	Specifies a username and password

Note that, if you do not supply a password, you may be prompted for one. If you want to avoid prompts during a batch run, either specify the password using the **-Y** switch, or use an account that does not require one.

This option is valid for the duration of the command it is used with only. TC will use your default account for commands that do not use this option, and automatically remembers these details between commands.

2.2 Commands

The following topics describe the commands available in the TC command-line utility in more detail. A few points should be noted, however:

Prompts and Confirmation

The TC command line only prompts for Comments and Username/Password, and only when necessary. *No confirmation of actions that could cause loss of local changes is requested from the user.*

For this reason, the built-in default options for all commands causes these commands to fail when local data may be overwritten. If you use command-line options or settings to override these defaults, **you may not be prompted for confirmation.**

Repository Connections

TC remembers the last repository you were connected to and the UserName/Password you used to connect to it. In addition, your position within a repository is also remembered between commands.

In order to switch to another repository, or to login as a different user for subsequent commands use the **Connect** command. To view the account you are using, use **WhoAmI**, or **ConnectInfo**.

Commands

You can specify other commands to use in place of the standard ones. This is useful if you prefer to use shortened commands, for example while typing. To specify alternatives to the builtin command names, modify or add the [Alternate] section in the TC.INI file.

For example, to specify UnCO as an alternate to UndoCheckOut, add the line **UnCO=UndoCheckOut** to the [Alternates] section of the INI file:

```
[Alternates]
UnCO=UndoCheckOut
```

Linux Users

Because of the differences in path specifications between Linux and Windows it is advisable to either use a [View](#) when checking files in or out, or to specify the destination folder specifically using the [-G](#) option.

Using a View is simpler in that all users can specify their own default working paths regardless of operating system, which allows you to simply select the view you are using based on the OS.

Default Settings

You can also specify defaults for the [-S](#), [-G](#), and [-C](#) command options using the TC.INI file. Information on setting the defaults for these options can be found under the [Command line Options](#) topic for the relative options.

2.2.1 About Team Coherence

Displays information about your copy of Team Coherence, including the version number and the copyright, legal and licensing notices.

2.2.2 Add

Adds new files into the Team Coherence Repository

Syntax

```
tc Add <local files> [-C] [-F] [-O] [-R] [-T] [-V] [-Y] [-?]
```

Note for Linux Users: When specifying the <local files> parameter, enclose each filespec in double quotes ("")

Examples

Adds file HELLO.PAS to the current folder:

```
tc Add C:\HELLO.PAS
```

Adds files TEST.PAS and MY LONG FILENAME.DPR to the current folder:

```
tc Add TEST.C "My long filename.H"
```

Adds all files in current directory to the current folder:

```
tc Add *
```

Adds all files in current directory and all subdirectories to the current project:

```
tc Add * -R
```

The following table describes the command-line options available with this command.

Option	Description
-C	Use the same comment for all added files
-F	Specify flags to modify the behaviour of Add
-O	Specify command output parameters
-R	Recursively add an entire directory tree, creating a folder list in Team Coherence
-T	Causes the command to terminate and return an error code when an error occurs.
-V	Assign a Version Label to the checked in file
-Y	Specify a Username and/or Password
-?, -H	Request online help for a command

Remarks

You can specify files on any drive and directory, and add that file to the current TC folder. To add a directory recursively, that is, to add all of its subdirectories and files, specify a directory instead of a file name and use the -R option.

Note This command supports Universal Naming Convention (UNC) names: it's not necessary to specify a drive. You can type, for example, \\COMPUTER\SHARE\FILE.TXT to add the FILE.TXT file from \\COMPUTER\SHARE.

2.2.3 AddConnection

Allows you to define the settings for a new connection to a repository.

Syntax

```
tc AddConnection <Name> <Description> <Host> <Port> [-D] [-K] [-O] [-U] [-X] [-?]
```

Examples

Adds a new connection called **Testing** to the repository on machine MCNServer that is listening on port 16000:

```
tc AddConnection Testing "Connection to MCNServer" MCNServer 16000
```

The following table describes the command-line options available with this command.

Option	Description
-D	Allows you to define Dial-Up Networking settings
-K	If the connection requires a specific encryption key to connect, use this parameter
-O	Specify command output parameters
-U	Forces communications between the client and server to be UUencoded.
-X	If the connection needs to pass through a SOCKS server, define parameters using this option.
-?, -H	Request online help for a command

Remarks

When defining a new connection, <Name>, <Description>, <Host>, and <Port> are required parameters and must be defined in that order.

2.2.4 AttachLabel

Attaches a Version Label to the latest revision of a file or to a specified revision.

Syntax

```
tc AttachLbl <Label> <TC Files> [-M] [-O] [-R] [-T] [-V] [-Y] [-?]
```

Examples

Attaches label **7.0.1.23** to the current revisions of the files in the current folder

```
tc AttachLbl 7.0.1.23 *
```

Recursively attaches label **My Version** to the current revision of all files in project //TestProject

```
tc AttachLbl "My Version" //Test Project -R
```

Attaches label Test to revision 1.3 of Hello.pas:

```
tc AttachLbl Test Hello.pas -VR1.3
```

The following table describes the command-line options available with this command.

Option	Description
-M	If an existing label with the same name exists, move it to the new revision.
-O	Specify command output parameters
-R	Recursively assign the label to the files contained in subfolders
-T	Causes the command to terminate and return an error code when an error occurs.
-V	Specify the revision to attach the label to
-Y	Specify a Username and/or Password
@<filename>	Specify a file containing the list of files to assign the label to
-?, -H	Request online help for a command

Remarks

Note This command supports Universal Naming Convention (UNC) names: it's not necessary to specify a drive. You can type, for example, \\COMPUTER\SHARE\FILE.TXT to add the FILE.TXT file from \\COMPUTER\SHARE.

2.2.5 CD

Sets the current Project or Folder

Syntax

```
tc cd [TC Object] [-O] [-T] [-Y] [-?]
```

Examples

Make project **Test Project** current:

```
tc cd "//Test Project"
```

Make folder **//Test Project/Source/Units** current:

```
tc cd "//Test Project/Source/Units"
```

Moves up to the parent of the current object:

```
tc cd ..
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-T	Causes the command to terminate and return an error code when an error occurs.
-Y	Specify a Username and/or Password
-?, -H	Request online help for a command

The **CD** command sets the current path in the repository. After you set the current path on the command line, most commands assume the current path if you simply specify a file name. For example, if you follow `tc Cd //Team Coherence/Source/Units` with `tc Get uMain.pas`, TC retrieves `//Team Coherence/Source/Units/uMain.pas`

Note Use the **CD** command without a path to find the current folder from the command line.

2.2.6 CheckIn

Updates Team Coherence with changes made to a checked out file and unlocks the archive.

Syntax

```
tc CheckIn <TC Files> [-C] [-F] [-G] [-LC] [-O] [-R] [-T] [-V] [-Y] [-?]
```

Examples

Checks in your changes to the file TEST.PAS:

```
tc CheckIn TEST.PAS
```

Checks in all files that you have checked out from the current folder:

```
tc CheckIn *
```

The following table describes the command-line options available with this command.

Option	Description
-C	Use the same comment for all checked in files
-F	Specify flags to modify the behaviour of CheckIn
-G	Specify the location of the workfile
-LC	Lock comments override comments entered on the commandline if they exist
-O	Specify command output parameters
-R	Recursively checkin a folder and its subfolders
-T	Causes the command to terminate and return an error code when an error occurs.
-V	Assign a Version Label to the checked in file
-Y	Specify a Username and/or Password
@<filename>	Specify a file containing the list of files to check in
-?, -H	Request online help for a command

Remarks

If you check in many files at once, TC will ask for a comment for each file. To use the same comment for all files, use the **-C** option.

It is possible to check a file in and create a new revision while keeping the file checked out for modification. To do this use the **-FL** option.

2.2.7 CheckOut

Extracts a file from the current folder to the current directory for the purposes of editing.

Syntax

```
tc CheckOut <TC Files> [-C] [-G] [-M] [-O] [-R] [-T] [-V] [-VA] [-W] [-Y] [-?]
```

Examples

Check out TEST.PAS from the folder //TestProject/Source:

```
tc CheckOut //TestProject/Source/TEST.PAS
```

Checks out all .pas and .dfm files from the current folder:

```
tc CheckOut *.pas *.dfm
```

Checks out all .pas files from the current folder and all subfolders:

```
tc CheckOut *.pas -r
```

The following table describes the command-line options available with this command.

Option	Description
-C	Use the same comment for all checked out files
-G	Specify the location of the workfile
-M	If assigning a Version Label (-VA), causes any existing label to be moved to the new revision.
-O	Specify command output parameters
-R	Recursively checkout a folder and its subfolders
-T	Causes the command to terminate and return an error code when an error occurs.
-V	Specify a revision or version to check out
-VA	Assign a Version Label to the checked out file
-W	If the local file is writable, causes it to be overwritten with the checked out copy
-Y	Specify a Username and/or Password
@<filename>	Specify a file containing the list of files to check out
-?, -H	Request online help for a command

Remarks

Check out comments are requested for all files included in the operation, by default. You can use the **-C** option to type just one comment, which will be applied to all files included in the operation.

2.2.8 Connect

Connects to a different repository.

Syntax

```
tc Connect <TC Repository> [-O] [-T] [-Y] [-?]
```

Examples

Connects to the repository called **Test Repository**:

```
tc Connect "Test Repository"
```

Connects to the repository called **Test Repository** and sets the default user to Ewan:

```
tc Connect "Test Repository" -YEwan,Ewanspassword
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-T	Causes the command to terminate and return an error code when an error occurs.
-Y	Specify a Username and/or Password
-?, -H	Request online help for a command

Remarks

When you connect to a different repository, it becomes the default for subsequent commands. If the username and password used for the current connection do not apply on the new connection, you will be prompted to enter a valid username and password. To avoid this prompt, use the **-Y** option.

Note that you must be connecting to a previously created repository connection. You can define connections using the Team Coherence Version Manager.

2.2.9 ConnectInfo

Displays the username and current repository connection.

Syntax

```
tc ConnectInfo [-O] [-Y] [-?]
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-Y	Specify a Username and/or Password
-?, -H	Request online help for a command

2.2.10 Create

Creates a new sub folder or project.

Syntax

```
tc Create <TC Object> [-O] [-S] [-T] [-Y] [-?]
```

Examples

Creates a folder called **New Folder** under the current project/folder:

```
tc Create "New Folder"
```

Creates a new project called **Test Project** and makes it current:

```
tc Create "//Test Project" -S
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-S	Enable Smart mode to select the object after creation
-T	Causes the command to terminate and return an error code when an error occurs.
-Y	Specify a Username and/or Password
-, -H	Request online help for a command

Remarks

By default, new folders are created with their working directory based on the parent object. To change the working directory used by a particular folder, use the WorkFolder command.

2.2.11 CreateLabel

Creates a new Version Label.

Syntax

```
tc CreateLbl LabelName [-O] [-C] [-P] [-T] [-Y] [-?]
```

Examples

Creates a new Version Label called **7.0.1.23**:

```
tc CreateLbl 7.0.1.23
```

Creates a new Version Label called **7.0.1.45** and associates it with the project **Team Coherence**:

```
tc CreateLbl 7.0.1.45 -P"Team Coherence"
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-C	Specify a comment for the new Label
-P	Associate the new Version Label with a Project
-T	Causes the command to terminate and return an error code when an error occurs.
-Y	Specify a Username and/or Password
-, -H	Request online help for a command

Remarks

You will be prompted for a comment for the new label. To suppress this during batch files see the -C command-line option.

Note you can also create labels on the fly during a check-in operation. To do this, see the -VCA option.

2.2.12 CreateView

Creates a new View.

Syntax

tc CreateView Name Description [Project List] [-B] [-O] [-T] [-V] [-Y] [-?]

Examples

Creates a new shared View based on projects **Team Coherence** and **SPac**, and on Promotion Label **Release**:

```
tc CreateView Release "Ready for release" "Team Coherence" "SPac" -VP"Release"
```

The following table describes the command-line options available with this command.

Option	Description
-B	Creates a shared View
-O	Specify command output parameters
-T	Causes the command to terminate and return an error code when an error occurs.
-V	Specify the basis of the View. -VL for a Version based View, -VP for a Promotional based View.
-Y	Specify a Username and/or Password
-?, -H	Request online help for a command

Remarks

Name and Description must be specified. if [Project List] is not defined, the View will apply to all projects.

2.2.13 Delete

Removes the selected objects from a repository and marks them as deleted.

Syntax

tc Delete <TC Items> [-O] [-T] [-Y] [-?]

Examples

Deletes all .pas and .dpr files from the current folder:

```
tc Delete *.pas *.dpr
```

Deletes the folder AFolder from the project //Test Project

```
tc Delete "//Test Project/AFolder"
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-T	Causes the command to terminate and return an error code when an error occurs.
-Y	Specify a Username and/or Password
-, -H	Request online help for a command

Remarks

There is NO confirmation for a delete operation. If a delete on a project or folder succeeds, all sub folders are also deleted. However the delete action causes the deleted object to be added to the Recycle Bin and they can be recovered using the Team Coherence Version Manager.

2.2.14 DeleteConnection

Deletes a specified connection from the connection list.

Syntax

```
tc DeleteConnection <Connection> [-O] [-?]
```

Examples

Deletes the connection named Team Coherence:

```
tc DeleteConnection "Team Coherence"
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-, -H	Request online help for a command

2.2.15 DeleteLabel

Deletes the specified Version Label.

Syntax

```
tc DeleteLbl LabelName [-O] [-T] [-Y] [-?]
```

Examples

Deletes the Version Label **7.0.1.23**:

```
tc DeleteLbl 7.0.1.23
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-T	Causes the command to terminate and return an error code when an error occurs.
-Y	Specify a Username and/or Password
-, -H	Request online help for a command

Remarks

There is NO confirmation for a delete operation. In the case of Version Labels, the label is permanently deleted and is removed from all files that contained it.

2.2.16 DeleteView

Deletes the specified View.

Syntax

tc DeleteView ViewName [-O] [-T] [-Y] [-?]

Examples

Deletes the View called Release:

```
tc DeleteView Release
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-T	Causes the command to terminate and return an error code when an error occurs.
-Y	Specify a Username and/or Password
-, -H	Request online help for a command

Remarks

There is NO confirmation for a delete operation. The View is permanently deleted.

2.2.17 DetachLabel

Detaches the specified label from the specified objects.

Syntax

tc DetachLbl <LabelName> <TC Items> [-O] [-R] [-T] [-V] [-Y] [-?]

Examples

Removes the Version Label **7.0.1.23** from all .pas and .dpr files in the current folder:

```
tc DetachLbl 7.0.1.23 *.pas *.dpr
```

Removes the label **My Label** from revision 1.4 of file HELLO.PAS:

```
tc DetachLbl "My Label" HELLO.PAS -VR1.4
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-R	Recursively detach the label from sub folders as well
-T	Causes the command to terminate and return an error code when an error occurs.
-V	Specify a specific revision
-Y	Specify a Username and/or Password
@<filename>	Specify a file containing the list of files to detach the label from
-, -H	Request online help for a command

2.2.18 Directory

Displays a list of the files and folders contained in the current project or folder, or the contents of a specific folder.

Syntax

```
tc Dir [TC Items] [-E] [-O] [-R] [-Y] [-?]
```

Examples

Displays the contents of the current folder:

```
tc Dir
```

Displays all .pas files in the folder //Test Project/Source:

```
tc Dir //Test Project/Source/*.pas
```

The following table describes the command-line options available with this command.

Option	Description
-E	Display extended information, including Lock information
-O	Specify command output parameters
-R	Lists files recursively
-Y	Specify a Username and/or Password
-, -H	Request online help for a command

2.2.19 Get

Gets a read-only copy of the specified files.

Syntax

```
tc Get <TC Files> [-E] [-G] [-M] [-O] [-R] [-T] [-V] [-VA] [-W] [-Y] [-?]
```

Examples

Get TEST.PAS from the folder //TestProject/Source:

```
tc Get //TestProject/Source/TEST.PAS
```

Gets all .pas and .dfm files from the current folder:

```
tc Get *.pas *.dfm
```

Gets all .pas files from the current folder and all subfolders:

```
tc Get *.pas -r
```

The following table describes the command-line options available with this command.

Option	Description
-E	Make the workfile writable after successful Get
-G	Specify the location of the workfile
-M	If assigning a Version Label (-VA), causes any existing label to be moved to the new revision.
-O	Specify command output parameters
-R	Recursively get a folder and its subfolders
-T	Causes the command to terminate and return an error code when an error occurs.
-V	Specify a revision or version to get
-VA	Assign a Version Label to the revision that was retrieved
-W	If the local file is writable, causes it to be overwritten with the checked out copy
-Y	Specify a Username and/or Password
@<filename>	Specify a file containing the list of files to get
-?, -H	Request online help for a command

Remarks

If the local copy of the file is not read-only the action will fail for that file. To override this default and to cause the local file to be overwritten with the new read-only copy, use the **-W** command-line option.

2.2.20 Help

Displays Help on using the TC Command-line in general, or on a specific command.

Syntax

```
tc Help [Command] [-O] [-Y] [-?]
```

Examples

Shows general help:

```
tc Help
```

Shows help on the CheckIn command:

```
tc Help CheckIn
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-Y	Specify a Username and/or Password
-?, -H	Request online help for a command

Remarks

Help for Team Coherence command-line commands is displayed in a Microsoft Windows Help file. You must be running Microsoft Windows for Help to be displayed - there is no separate command-line help.

2.2.21 HistoryReport

Generates a History report for the specified files. Reports are output in html to the named file.

Syntax

```
tc HistoryReport <TC Files> -AO<filename> [-ADF] [-ADT] [-I] [-R] [-VF] [-VT] [-VI]
```

Examples

Generate the history report for all files in the current folder, including the Name and Comments, and Timestamp columns:

```
tc HistoryReport * -AO"c:\report.htm" -I7
```

Generate a history report for all .pas files, including in all subfolders, and including revisions between version "1.2.3" and "1.3.4":

```
tc HistoryReport *.pas -R -AO"c:\report.htm" -I127 -VF"1.2.3" -VT"1.3.4"
```

The following table describes the command-line options available with this command.

Option	Description
-ADF	Include all revisions later than the passed date.
-ADT	Include all revisions prior to and including the passed date.
-I	Columns to include in the report.
-R	Recurse through subfolders.
-VF	Include all revisions after the one this label is attached to.
-VT	Include revisions up until the one this label is attached to.
-VI	If passed, does not display the 'from' revision.
-Y	Specify a Username and/or Password
-?, -H	Request online help for a command

Remarks

Dates and Version Labels should be enclosed in quotes if there are any spaces. Dates are converted using the system format of the client machine.

2.2.22 ListConnections

Lists the all the available connections.

Syntax

```
tc ListConnections [-O] [-Y] [-?]
```

Examples

Displays basic information for the list of defined connections available from this machine.

```
tc ListConnections
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-?, -H	Request online help for a command

2.2.23 ListLockedFiles

Lists the full path of the files locked by the current user.

Syntax

```
tc ListLockedFiles [-O] [-Y] [-?]
```

Examples

Displays the list of files locked by the current user

```
tc Dir
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-Y	Specify a Username and/or Password
-, -H	Request online help for a command

2.2.24 Promote

Promotes the specified files to the next Promotion Level. This command will promote the tip revision (as defined by the current View) to the next higher Promotion Level.

Syntax

```
tc Promote <TC Files> [-O] [-R] [-T] [-Y] [-?]
```

Examples

Promotes all files in the current folder, and its subfolders, to the next higher Promotion Level

```
tc Promote * -R
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-R	Recursively Promote the files contained in subfolders
-T	Causes the command to terminate and return an error code when an error occurs.
-Y	Specify a Username and/or Password
@<filename>	Specify a file containing the list of files to promote
-, -H	Request online help for a command

Remarks

Both the Tip Revision and the next higher Promotion Level are defined by the currently selected View. If the current View is based on a Version Label, the revision that has the Version Label attached will be promoted to the **first** Promotion Level. If the current View is based on a Promotion Level, the revision that is currently at that level will be promoted to the **next higher** Promotion Level.

If the **<default>** View is current, or the current View is based on neither a Version Label or Promotion Level, the tip revision of each file will be promoted.

2.2.25 Rename

Renames a folder or project.

Syntax

```
tc Rename OldName NewName [-O] [-T] [-Y] [-?]
```

Examples

Rename the sub folder called **OldName** to one called **Real Name**:

```
tc Rename OldName "Real Name"
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-T	Causes the command to terminate and return an error code when an error occurs.
-Y	Specify a Username and/or Password
-, -H	Request online help for a command

2.2.26 RenameLabel

Renames a Version Label

Syntax

```
tc RenameLbl OldName NewName [-C] [-O] [-T] [-Y] [-?]
```

Examples

Rename the Label called **7.0.1.23** to one called **New Name**:

```
tc Rename 7.0.1.23 "New Name"
```

The following table describes the command-line options available with this command.

Option	Description
-C	Modify the comments assigned to the label
-O	Specify command output parameters
-T	Causes the command to terminate and return an error code when an error occurs.
-Y	Specify a Username and/or Password
-, -H	Request online help for a command

2.2.27 SetView

Selects a different view and makes it current.

Syntax

```
tc SetView ViewName [-O] [-T] [-Y] [-?]
```

Examples

Selects the view named QA:

```
tc SetView QA
```

Selects the default View:

```
tc SetView "<default>"
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-T	Causes the command to terminate and return an error code when an error occurs.
-Y	Specify a Username and/or Password
-?, -H	Request online help for a command

Remarks

For more information on Views, see the main Team Coherence help file.

2.2.28 Sync

Similar to the [Get](#) command, Sync makes sure that the local copy of your source is as up to date as possible. Sync updates all files that are marked as Out-of-Date (checked in by others since the last Get) or missing. This is faster than running a Get since local files that are checked out to you, or do not have a new revision in the repository, are ignored.

Syntax

```
tc Sync [-O] [-R] [-T] [-Y] [-?]
```

Examples

Synchronise the current folder and all subfolders:

```
tc Sync -R
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-R	Recursively get a folder and its subfolders
-T	Causes the command to terminate and return an error code when an error occurs.
-Y	Specify a Username and/or Password
-?, -H	Request online help for a command

Remarks

The action is carried out on the current Team Coherence folder (use `TC CD` to check this) and uses the user working folder to locate the local source files (equivalent to the `-GW` parameter in a `Get`).

2.2.29 Undo CheckOut

Reverses a checkout action.

Syntax

```
tc UndoCheckOut <TC Items> [-O] [-T] [-Y] [-?]
```

Examples

Undo a checkout on Hello.pas in the current folder:

```
tc UndoCheckOut Hello.pas
```

Undo a checkout on all file in the folder //Test Project/Source/Units

```
tc UndoCheckout //Test Project/Source/Units/*
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-T	Causes the command to terminate and return an error code when an error occurs.
-Y	Specify a Username and/or Password
@<filename>	Specify a file containing the list of files to undo the checkout on
-?, -H	Request online help for a command

Remarks

This command effectively unlocks the archive and performs a `Get`. If changes were made to the local file, these changes will be lost.

2.2.30 UpdateConnection

Allows you to modify the settings for an existing repository connection.

Syntax

```
tc UpdateConnection <Name> [-D] [-K] [-N] [-O] [-U] [-X] [-?]
```

Examples

Modifies the connection called **Testing** and changes the Host to localhost and the Port to 2000:

```
tc UpdateConnection Testing -NHlocalhost -NP2000
```

The following table describes the command-line options available with this command.

Option	Description
-D	Allows you to define Dial-Up Networking settings
-K	If the connection requires a specific encryption key to connect, use this parameter
-O	Specify command output parameters
-N	Allows you to modify the Description, Host and Port settings for the connection.
-U	Forces communications between the client and server to be UUencoded.
-X	If the connection needs to pass through a SOCKS server, define parameters using this option.
-?, -H	Request online help for a command

2.2.31 UpdateView

Modifies an existing View.

Syntax

```
tc UpdateView Name [Project List] [-B] [-O] [-T] [-V] [-Y] [-?]
```

Examples

Updates an existing View, and bases it on project **Team Coherence** and on Version Label **7.1.2.90**:

```
tc UpdateView "Version 7" "Team Coherence" -VL"7.1.2.90"
```

The following table describes the command-line options available with this command.

Option	Description
-B	Makes the View Shared
-O	Specify command output parameters
-T	Causes the command to terminate and return an error code when an error occurs.
-V	Specify the basis of the View. -VL for a Version based View, -VP for a Promotional based View.
-Y	Specify a Username and/or Password
-?, -H	Request online help for a command

Remarks

Name must be specified. if [Project List] is not defined, the View will apply to all projects.

2.2.32 Views

Displays the currently defined Views

Syntax

tc Views [-O] [-Y] [-?]

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-Y	Specify a Username and/or Password
-?, -H	Request online help for a command

Remarks

The currently active View is highlighted with an asterisk. For more information on Views, see the main Team Coherence help file.

2.2.33 Whoami

Displays the username you are logged in under.

Syntax

tc ConnectInfo [-O] [-Y] [-?]

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-Y	Specify a Username and/or Password
-?, -H	Request online help for a command

2.2.34 WorkFolder

Changes the default working directory for the specified folder.

Syntax

tc WorkFold <TC Folder> <Local Path> [-O] [-T] [-Y] [-?]

Sets the //Website/html folder working directory to c:\External Web\Website\html:

```
tc Workfold //WEBSITE/Html "c:\External Web\Website\html"
```

The following table describes the command-line options available with this command.

Option	Description
-O	Specify command output parameters
-T	Causes the command to terminate and return an error code when an error occurs.
-Y	Specify a Username and/or Password
-?, -H	Request online help for a command

Remarks

This command changes the default working directory to the new path for the current user. It does not affect the working directory for other users.

Index

- - -

-? Command-line Option 8

- A -

About 17
Add 18
AttachLabel 19

- - -

-C Command-line Option 9

- C -

CD 20
CheckIn 21
CheckOut 22
Command-line Options 8
Commands 17
Connect 22
ConnectInfo 23
Create 23
CreateLabel 24

- D -

Delete 25
DeleteLabel 26
DetachLabel 27
Directory 28

- - -

-F Command-line Option 10
-G Command-line Option 11

- G -

Get 29

- H -

Help 29

- - -

-M Command-line Option 12
-O Command-line Option 12
-R Command-line Option 13

- R -

Rename 33
RenameLabel 33

- - -

-S Command-line Option 13

- S -

SetView 34

- U -

UndoCheckOut 35

- - -

-V Command-line Option 15

- V -

Views 37

- - -

-W Command-line Option 16

- W -

WhoAml 37
WorkFolder 38

- - -

-Y Command-line Option 16

