Concurrent Versions System
The open standard for version control

CVS | CVS Communication | CVS Documentation | CVS Downloads | Other Local Projects

Domain Navigation
? Domain Home
? CVS Manual
? CVS FAQ
? Hosted Projects
? Hosted Tools
? Site Help
? SourceCast FAQ
? Site Help
? Feedback

Login:
Password:
Login
? Not a member? Register Now

Search:
Go

CVS Quick Reference

A CVS command looks like:

```
cvs [ global_options ] command [ command_options ] [ command_
```

Global options:

`--allow-root=`*rootdir*
: Specify legal CVSROOT directory (server only) (not in CVS 1.9 and older).

`-a`
: Authenticate all communication (client only) (not in CVS 1.9 and older).

`-b`
: Specify RCS location (CVS 1.9 and older).

`-d` *root*
: Specify the CVSROOT.

`-e` *editor*
: Edit messages with *editor*.

`-f`
: Do not read the `~/.cvsrc' file.

`-H`
`--help`
: Print a help message.

`-l`
: Do not log in CVSROOT/history file.

`-n`
: Do not change any files.

`-Q`
: Be really quiet.

`-q`
: Be somewhat quiet.

`-r`
: Make new working files read-only.

`-s` *variable=value*
: Set a user variable.

`-T` *tempdir*
: Put temporary files in *tempdir*.

`-t`
: Trace CVS execution.

`-v`
`--version`
: Display version and copyright information for CVS.

`-w`
: Make new working files read-write.

`-x`
: Encrypt all communication (client only).

`-z` *gzip-level*
: Set the compression level (client only).

Feedback

Keyword expansion modes

```
-kkv  $Id: ref.html,v 1.10 2001/06/15 02:18:18 dprice Exp $
-kkvl $Id: ref.html,v 1.10 2001/06/15 02:18:18 dprice Exp $
-kk   $Id: ref.html,v 1.10 2001/06/15 02:18:18 dprice Exp $
-kv   info-ref.html,v 1.1 1999/04/14 19:04:02 kingdon Exp
-ko   no expansion
-kb   no expansion, file is binary
```

Keywords

```
$Author: dprice $
$Date: 2001/06/15 02:18:18 $
$Header: /usr/local/tigris/data/helm/cvs/repository/www/www/d
$Id: ref.html,v 1.10 2001/06/15 02:18:18 dprice Exp $
$Locker:  $
$Name:  $
$RCSfile: ref.html,v $
$Revision: 1.10 $
$Source: /usr/local/tigris/data/helm/cvs/repository/www/www/d
$State: Exp $
$Log: ref.html,v $
Revision 1.10  2001/06/15 02:18:18  dprice
Tidied.

Revision 1.1  1999/04/14 19:04:02  kingdon
By popular demand, want to put the CVS Reference Card on-line
```

Commands, command options, and command

`add [`*options*`] [`*files...*`]`
: Add a new file/directory.

`-k` *kflag*
: Set keyword expansion.

`-m` *msg*
: Set file description.

`admin [options] [files...]`
Administration of history files in the repository.

`-b[rev]`
Set default branch.
`-cstring`
Set comment leader.
`-ksubst`
Set keyword substitution.
`-l[rev]`
Lock revision *rev*, or latest revision.
`-mrev:msg`
Replace the log message of revision *rev* with *msg*.
`-orange`
Delete revisions from the repository.
`-q`
Run quietly; do not print diagnostics.
`-sstate[:rev]`
Set the state.
`-t`
Set file description from standard input.
`-tfile`
Set file description from *file*.
`-t-string`
Set file description to *string*.
`-u[rev]`
Unlock revision *rev*, or latest revision.

`annotate [options] [files...]`
Show last revision where each line was modified.

`-D date`
Annotate the most recent revision no later than *date*.
`-f`
Use head revision if tag/date not found.
`-l`
Local; run only in current working directory.
`-R`
Operate recursively (default).
`-r tag`
Annotate revision *tag*.

`checkout [options] modules...`
Get a copy of the sources.

`-A`
Reset any sticky tags/date/options.
`-c`
Output the module database.
`-D date`
Check out revisions as of *date* (is sticky).
`-d dir`
Check out into *dir*.
`-f`
Use head revision if tag/date not found.
`-j rev`
Merge in changes.
`-k kflag`
Use *kflag* keyword expansion.
`-l`
Local; run only in current working directory.
`-N`
Don't "shorten" module paths if -d specified.
`-n`

Do not run module program (if any).
`-P`
Prune empty directories.
`-p`
Check out files to standard output (avoids stickiness).
`-R`
Operate recursively (default).
`-r tag`
Checkout revision *tag* (is sticky).
`-s`
Like -c, but include module status.

`commit [options] [files...]`
Check changes into the repository.

`-F file`
Read log message from *file*.
`-f`
Force the file to be committed; disables recursion.
`-l`
Local; run only in current working directory.
`-m msg`
Use *msg* as log message.
`-n`
Do not run module program (if any).
`-R`
Operate recursively (default).
`-r rev`
Commit to *rev*.

`diff [options] [files...]`
Show differences between revisions. In addition to the options shown below
example `-c' for context diffs.

`-D date1`
Diff revision for date against working file.
`-D date2`
Diff *rev1/date1* against *date2*.
`-l`
Local; run only in current working directory.
`-N`
Include diffs for added and removed files.
`-R`
Operate recursively (default).
`-r rev1`
Diff revision for *rev1* against working file.
`-r rev2`
Diff *rev1/date1* against *rev2*.

`edit [options] [files...]`
Get ready to edit a watched file.

`-a actions`
Specify actions for temporary watch, where *actions* is `edit`, `unedit`
`-l`
Local; run only in current working directory.
`-R`
Operate recursively (default).

`editors [options] [files...]`
See who is editing a watched file.

`-l`
Local; run only in current working directory.

`-R`
Operate recursively (default).

`export [options] modules...`
Export files from CVS.

`-D date`
Check out revisions as of *date*.
`-d dir`
Check out into *dir*.
`-f`
Use head revision if tag/date not found.
`-k kflag`
Use *kflag* keyword expansion.
`-l`
Local; run only in current working directory.
`-N`
Don't "shorten" module paths if -d specified.
`-n`
Do not run module program (if any).
`-P`
Prune empty directories.
`-R`
Operate recursively (default).
`-r tag`
Checkout revision *tag*.

`history [options] [files...]`
Show repository access history.

`-a`
All users (default is self).
`-b str`
Back to record with *str* in module/file/repos field.
`-c`
Report on committed (modified) files.
`-D date`
Since *date*.
`-e`
Report on all record types.
`-l`
Last modified (committed or modified report).
`-m module`
Report on *module* (repeatable).
`-n module`
In *module*.
`-o`
Report on checked out modules.
`-r rev`
Since revision *rev*.
`-T`
Produce report on all TAGs.
`-t tag`
Since tag record placed in history file (by anyone).
`-u user`
For user *user* (repeatable).
`-w`
Working directory must match.
`-x types`
Report on *types*, one or more of `TOEFWUCGMAR`.
`-z zone`
Output for time zone *zone*.

`import [options] repository vendor-tag release-tags...`

Import files into CVS, using vendor branches.

`-b bra`
Import to vendor branch *bra*.
`-d`
Use the file's modification time as the time of import.
`-k kflag`
Set default keyword substitution mode.
`-m msg`
Use *msg* for log message.
`-I ign`
More files to ignore (! to reset).
`-W spec`
More wrappers.

`init`
Create a CVS repository if it doesn't exist.
`log [options] [files...]`
Print out history information for files.

`-b`
Only list revisions on the default branch.
`-d dates`
Specify dates (*d1*<*d2* for range, *d* for latest before).
`-h`
Only print header.
`-l`
Local; run only in current working directory.
`-N`
Do not list tags.
`-R`
Only print name of RCS file.
`-rrevs`
Only list revisions *revs*.
`-s states`
Only list revisions with specified states.
`-t`
Only print header and descriptive text.
`-wlogins`
Only list revisions checked in by specified logins.

`login`
Prompt for password for authenticating server.
`logout`
Remove stored password for authenticating server.
`rdiff [options] modules...`
Show differences between releases.

`-c`
Context diff output format (default).
`-D date`
Select revisions based on *date*.
`-f`
Use head revision if tag/date not found.
`-l`
Local; run only in current working directory.
`-R`
Operate recursively (default).
`-r rev`
Select revisions based on *rev*.
`-s`
Short patch - one liner per file.
`-t`
Top two diffs - last change made to the file.

`-u`
: Unidiff output format.

`-V` *vers*
: Use RCS Version *vers* for keyword expansion (obsolete).

`release [`*options*`] `*directory*
Indicate that a directory is no longer in use.

`-d`
: Delete the given directory.

`remove [`*options*`] [`*files...*`]`
Remove an entry from the repository.

`-f`
: Delete the file before removing it.

`-l`
: Local; run only in current working directory.

`-R`
: Operate recursively (default).

`rtag [`*options*`] `*tag modules...*
Add a symbolic tag to a module.

`-a`
: Clear tag from removed files that would not otherwise be tagged.

`-b`
: Create a branch named *tag*.

`-D` *date*
: Tag revisions as of *date*.

`-d`
: Delete *tag*.

`-F`
: Move *tag* if it already exists.

`-f`
: Force a head revision match if tag/date not found.

`-l`
: Local; run only in current working directory.

`-n`
: No execution of tag program.

`-R`
: Operate recursively (default).

`-r` *rev*
: Tag existing tag *rev*.

`status [`*options*`] `*files...*
Display status information in a working directory.

`-l`
: Local; run only in current working directory.

`-R`
: Operate recursively (default).

`-v`
: Include tag information for file.

`tag [`*options*`] `*tag*` [`*files...*`]`
Add a symbolic tag to checked out version of files.

`-b`
: Create a branch named *tag*.

`-c`
: Check that working files are unmodified.

`-D` *date*

: Tag revisions as of *date*.

`-d`
: Delete *tag*.

`-F`
: Move *tag* if it already exists.

`-f`
: Force a head revision match if tag/date not found.

`-l`
: Local; run only in current working directory.

`-R`
: Operate recursively (default).

`-r` *rev*
: Tag existing tag *rev*.

`unedit [`*options*`] [`*files...*`]`
Undo an edit command.

`-a` *actions*
: Specify actions for temporary watch, where *actions* is `edit`, `unedit`

`-l`
: Local; run only in current working directory.

`-R`
: Operate recursively (default).

`update [`*options*`] [`*files...*`]`
Bring work tree in sync with repository.

`-A`
: Reset any sticky tags/date/options.

`-D` *date*
: Check out revisions as of *date* (is sticky).

`-d`
: Create directories.

`-f`
: Use head revision if tag/date not found.

`-I` *ign*
: More files to ignore (! to reset).

`-j` *rev*
: Merge in changes.

`-k` *kflag*
: Use *kflag* keyword expansion.

`-l`
: Local; run only in current working directory.

`-P`
: Prune empty directories.

`-p`
: Check out files to standard output (avoids stickiness).

`-R`
: Operate recursively (default).

`-r` *tag*
: Checkout revision *tag* (is sticky).

`-W` *spec*
: More wrappers.

`watch [on|off|add|remove] [`*options*`] [`*files...*`]`
on/off: turn on/off read-only checkouts of files. add/remove: add or remove

`-a` *actions*
: Specify actions for temporary watch, where *actions* is `edit`, `unedit`

`-l`
: Local; run only in current working directory.

`-R`
: Operate recursively (default).

`watchers [options] [files...]`
See who is watching a file.

`-l`
Local; run only in current working directory.

`-R`
Operate recursively (default).

Copyright © 1999-2002 CollabNet, Inc. CollabNet and SourceCast are tradema
All other trademarks or registered trademarks are the property of their respect