Subversion Quick Reference Card

Subversion is a version control system that is a replacement for CVS. It has most of CVS's features. Generally, Subversion's interface to a particular feature is similar to CVS's, except where there's a compelling reason to do otherwise.

Quick start
svnadmin create /var/svnroot
svn import LocalDir file:///var/svnroot/ProjectName
svn checkout file:///var/svnroot/ProjectName
svn help [command]

Subversion URLs
file:/// Direct repository access (on local disk).
https:// Same as http://, but with SSL encryption.
svn:// Access via custom protocol to an svnserve server.
svn+ssh:// Same as svn://, but through an SSH tunnel.

svn subcommands
add Adds files and directories.
blame (praise, annotate, an) Shows author and revision information in-line for the specified files or URLs.
cat Outputs the contents of the specified files or URLs.
(checkout) Checks out a working copy from a repository.
(cleanup) Recursively clean up the working copy.
commit (c) Sends changes from your working copy to the repository.
copy (p) Copy a file or directory in a working copy or in the repository.
delete (del, remove, rm) Delete an item from a working copy or the repository.
diff (di) Display the differences between two paths.
export Prints help text.
help (help) Recursively commit a copy of local dir into a repository.
info Print information about PATHs.
(list) List directory entries in the repository.
log Displays commit log messages.

merge Apply the differences between two sources to a working copy path.
mkdir Create a new directory under version control.
move (mv, rename, ren) Move a file or directory.
(propel, pe) Remove a property from an item.
(propedit, pedit) Edit the property of one or more items under version control.
(propget, pg) Prints the value of a property.
(propiset, pi) Lists all properties.
(propset, ps) Sets a property on files, directories, or revisions.
resolved Remove conflicted state on working copy files or directories.
revert Undo all local edits.
status (stat, st) Print the status of working copy files and directories.
switch (sw) Update working copy to a different URL.
update (up) Updates your working copy.

svn switches
#confdir DIR
Read configuration from DIR instead of ~/.subversion.
#d DCMO
Use external program CMD for generating diff output instead of internal diff engine.
#diffcmd CMD
Use external program CMD for merging files.
#dryrun Run the command without changing anything.
#editorcmd CMD
Use external program CMD for editing files.
#encoding ENC
Instructs Subversion to use encoding ENC to store log messages.
#extensions ARG (ex)
Additional arguments for external diff, eg: svn #d --diffcmd "diff -e"
#file FILE (f)
Use the contents of FILE as an argument for a given subcommand.
#force Forces a particular subcommand to run.
#force=log Forces a suspicious parameter passed to the #message options to be accepted as valid.
#help (h or @)
Prints help for a given command or general help text.

#ignore=ancestry Ignore ancestry when calculating differences (rely on path contents alone).
#incremental Prints output in a format suitable for concatenation.
#message (em) MSG Specify a commit message MSG.
#mount=cache Prevents caching of authentication information.
#auto=props Disable auto=props, overriding the enable=auto=props directive in the config file.
#revidifdeleted Prevents Subversion from printing differences for deleted files.
#ignore Shows files in the status listing that would normally be omitted since they match a pattern in the svn:ignore property.
#noninteractive Prevents prompting for authentication information.
#no=recursive (#r)
Stops a subcommand from recursing into subdirectories.
#notice=ancestry Pay attention to ancestry when calculating differences.
#old ARG Uses ARG as the older target.
#password PASS Password for authentication.
#quiet (q)
Print only essential information while performing an operation.
#recursive (#r)
Makes a subcommand recurse into subdirectories.
#relocate FROM TO [PATH...]
Used with the svn switch subcommand, changes the location of the repository that your working copy references.
#revision REV (#r)
Supply a revision REV (or range of revisions) for a particular operation.
#revprop Operates on a revision property instead of a Subversion property specific to a file or directory (requires #revision switch).
#showupdates (#u)
Causes the client to display information about which files in your working copy are out-of-date.
## svnlook subcommands

- **author**
  Prints the author.

- **cat**
  Prints the contents of a path.

- **changed**
  Prints the paths that were changed.

- **date**
  Prints the date stamp.

- **diff**
  Prints differences of changed paths and properties.

- **dirs**
  Prints the directories that were themselves changed.

- **help**
  Print information about the history of a path in the repository.

- **info**
  Print the author, date stamp, log message size, and log message.

- **log**
  Print the log message.

- **prop**
  Print the names and values of versioned path and directory properties.

- **tree**
  Print the tree.

- **youngest**
  Print the youngest revision number.

## svnlook switches

- **-D**
  Prevents svnlook from printing differences for deleted paths.

- **-r**
  Specify a particular revision number that you wish to examine.

- **-t**
  Specify a particular transaction id TID that you wish to examine.

- **-A**
  Show the plesystem node revision IDs for each path in the plesystem tree.

## File status

- **U**
  File was updated.

- **A**
  File was added.

- **D**
  File was deleted.

- **R**
  File was replaced.

- **G**
  File was merged.

- **C**
  Conlicting changes.

- **?**
  Resource is not under version control.

- **!**
  Resource is missing or incomplete (removed by another tool than Subversion).

## Special properties

- **svn:executable**
  Executable file permission.

---

```
ioconf
List of file patterns which certain Subversion operations will ignore. Full list may be obtained by running svn status -n.<

-eol
Possible values are: \n, CRLF, LF, CR.

event
Instructions for Subversion to populate a versioned directory with one or more other checked-out Subversion working copies.

date
List of keywords that will be substituted during commit:
$Date$ date of the last modification
$Rev$ revision number
$Author$ the last user who changed the file
$URL$ full URL to the latest version of the file in the repository
$Id$ compressed combination of keywords above
```