

**NAME**

**svn switch** — Update the working copy to a different URL

**SYNOPSIS**

```
svn switch (sw) [ -Nq ] [ --accept ] [ --config-dir DIR ] [ --config-option OPT ]
    [ --depth DEP ] [ --diff3-cmd ] [ --force ] [ --ignore-externals ]
    [ --non-recursive ] [ --no-auth-cache ] [ --non-interactive ]
    [ --password PWD ] [ --quiet ] [ {-r | --revision} REV ] [ --relocate ]
    [ --set-depth ] [ --trust-server-cert ] [ --username USR ]
    URL [ @PEGREV ] [ PATH ... ]

svn switch --relocate FROM TO [ PATH ... ]
```

**DESCRIPTION**

The first form updates the working copy to mirror a new URL within the repository. This behavior is similar to **svn update**, and is the way to move a working copy to a branch or tag within the same repository. If specified, *PEGREV* determines in which revision the target is first looked up.

If **--force** is used, unversioned obstructing paths in the working copy do not automatically cause a failure if the switch attempts to add the same path. If the obstructing path is the same type (file or directory) as the corresponding path in the repository it becomes versioned but its contents are left 'as-is' in the working copy. This means that an obstructing directory's unversioned children may also obstruct and become versioned. For files, any content differences between the obstruction and the repository are treated like a local modification to the working copy. All properties from the repository are applied to the obstructing path.

Use the **--set-depth** option to set a new working copy depth on the targets of this operation.

The second form rewrites working copy URL metadata to reflect a syntactic change only. This is used when repository's root URL changes (such as a scheme or hostname change) but your working copy still reflects the same directory within the same repository. See also 'svn help update' for a list of possible characters reporting the action taken.

**OPTIONS**

|                           |  |
|---------------------------|--|
| <b>--accept</b>           | specify automatic conflict resolution source, one of:<br>base<br>working<br>mine-conflict<br>theirs-conflict<br>mine-full<br>theirs-full<br>edit<br>launch |
| <b>--depth DEP</b>        | limit operation by depth <i>ARG</i> , one of:<br>empty<br>files<br>immediates<br>infinity  |
| <b>--diff3-cmd</b>        | use <i>ARG</i> as merge command  |
| <b>--force</b>            | force operation to run   |
| <b>--ignore-externals</b> | ignore externals definitions   |
| <b>-N --non-recursive</b> | obsolete; try <b>--depth=files</b> or <b>--depth=immediates</b>  |

**-q --quiet** print nothing, or only summary information  
**-r --revision** *REV* the revision argument *REV* can be one of:  
     *NUMBER* revision number  
     ‘{’ *DATE* ‘}’ revision at start of the date  
     HEAD latest in repository  
     BASE base rev of item's working copy  
     COMMITTED last commit at or before BASE  
     PREV revision just before COMMITTED  
 Some commands also take an *ARG1* : *ARG2* range.  
**--relocate** relocate via URL-rewriting  
**--set-depth** set new working copy depth to *ARG*  
     exclude  
     empty  
     files  
     immediates  
     infinity

## GLOBAL OPTIONS

**--config-dir** *DIR* read user configuration files from directory *DIR*  
**--config-option** *OPT* set user configuration option *OPT* in the format:  
     FILE:SECTION:OPTION=[VALUE]  
 For example:  
     servers:global:http-library=serf  
**--no-auth-cache** do not cache authentication tokens  
**--non-interactive** do no interactive prompting  
**--password** *PWD* specify a password *PWD*  
**--trust-server-cert** accept unknown SSL server certificates without prompting (but only with  
     **--non-interactive**)  
**--username** *USR* specify a username *USR*