

## CvsQuickRef

### Table of contents

---

## Quick Reference Sheet - CVS For Web Projects

### LOGIN

Sign on to the CVS server so you have permission to check in and check out web sites. Run cvs login from your development folder.

```
cvs login
cd /development/
cvs login
```

### IMPORT

Send a whole new site up to the server. cvs import is **not used very often**. Run cvs import from inside the root htdocs folder on the server/site to be imported.

```
cvs import -m'message' clientdomain vendorname start
telnet www.sandiegozoo.org
cd htdocs/; cvs login
cvs import -m"import the zoo" sandiegozoo.org interactivate start
```

### CHECKOUT

Grab a working copy of a site (or any part of a site) and put it onto your desktop computer. cvs checkout gets the most recent version of the site by default, but can also be used to checkout previous versions of a site or file. Run cvs checkout from your development folder. A subfolder will be created to contain each site that you checkout.

```
cvs checkout clientdomain
cd /development/
cvs checkout kawanhee.org/index.html
```

### UPDATE

Freshen your working copy of a site. Adds any new files from the CVS server to your working copy, tries to automatically merge changes. You can update your working copy to turn it into any previous version of the site. cvs update defaults to the latest version. Run cvs update from inside the folder you would like to update.

```
cvs update
cd /development/imagebase/
cvs -q update -rdwiggins -dP
```

### ADD

Schedule a list of newly created files that exist in your working copy to be added to the CVS site server the next time you commit.

```
cvs add file1 file2...
cd /development/calstrawberry.com/
cvs add new1.html olddir/new2.html newdir
```

### REMOVE

Schedule files to be removed from the CVS server the next time you commit this site. cvs remove can be

run from any location inside your working copy of a site.

```
cv$ remove file1 file2...
cd /development/padres.com/
cv$ remove -f old1.html dir/old2.html olddir
```

## COMMIT

Send changes made to files in your working copy up to the CVS server. Run `cv$ commit` from your working copy folder.

```
cv$ commit file1 file2...
cd /development/californiamart.com/
cv$ commit
```

## RELEASE

Sign out with the CVS server, make sure there are no loose ends hidden within your working copy. Run this command from your development folder.

```
cv$ release clientdomain
cd /development/
cv$ release workforce.org/
```