

CVS Summary

The first few times you use CVS, ask for help. If you think you have messed up the CVS repository in any way, you should not be shy of reporting it.

Logging on:

```
cvs login
```

(You will need to supply password)

Creating a branch from the current live or working version:

This will have to be done before you can work on files.

You will need a *branch name from which to branch* (i.e. the live branch name) – *original_branch* - and a *module name*. You will also have to specify a new working branch name – *new_branch*.

- Move to subfolder under which code is to appear.
- **cv**s checkout -r *original_branch* *module_name*
- **cv**s co -r *new_branch* *module_name*

e.g with a branch name of **RC_Training_19_03_2012** and a module name **training/appserver**:

```
cvs checkout -r RC_Training_19_03_2012 training/appserver  
cvs co -r B_rbowles_scribble training/server
```

This gives a new branch name of **B_rbowles_scribble**.

Updating files in working subfolder:

This copies files that have changed from the repository to your working folder. You should always do this before making changes to a file to ensure that you are working on the latest version of the code. It is also a good idea to update before carrying out a merge operation.

- Move to the subfolder that requires the updated files
- **cv**s update -dP

The -d option creates new subfolders if they have been added to the branch. The P option purges old/unnecessary subfolders. Always update before committing a file to the repository. The keyword **update** can be abbreviated to **upd**

Files containing conflicts cannot be run as TCL programs as the additional symbols introduced by CVS causes the program to abort. Load the file into a text editor, and remove the <<<<<<<<<<< line, the ===== line, the >>>>>>>>>>> line and one of the conflicting pieces of code.

It is a good idea to consult your colleagues before removing pieces of code that they have put in!

Merging into the live branch

When you are sure that your code is ready to go live, you merge it into the live branch as follows:

```
cvs update -j working-branch-name
```

It is a good idea to use **cv**s **stat** before doing a merge to ensure that you are about to merge from the correct branch. When you have carried out a merge, commit the files with the following standard message:

```
cvs commit -m "Merge of branch working-branch-name"
```

(Don't specify a file name so that the same message will be given to all merged files).

Check the Ladbrokes CVS Convention:

<https://twiki.openbet/bin/view/Customers/LBRCVSTaggingSystem>