Environnement Cvs

Table des matières

A priori cvs repère les modifications en se basant sur 3 lignes inchangées au dessus et au dessous. `cvs init` permet de créer (suite à l'installation) le fichier *$CVSROOT/CVSROOT*.

1 Import

Voici comment insérer un nouveau projet dans le serveur cvs.
*cvs import -m "*TOPO*"* DIR VENDOR_TAG RELEASE_TAG *// met . en conf*
Exemple :

```
nroche@narval:~/src/narvali$ export CVSROOT=/cvsroot
nroche@narval:~/src/narvali$ export CVSEDITOR=/usr/bin/vi
nroche@narval:~/src/narvali$ make clean
nroche@narval:~/src/narvali$ cvs import -m "mon intranet" narvali/ nroche v1
```

1.1 Ménage

Supprimer le répertoire dans $CVSROOT et ça roule !

2 Checkout

cvs checkout DIR
Exemples :

2.1 Local

```
nroche@narval:~/cvs$ export CVSROOT=/cvsroot
nroche@narval:~/cvs$ export CVSEDITOR=/usr/bin/vi
nroche@narval:~/cvs$ cvs checkout narvali
```

2.2 Distant

```
nroche@planet-gis:~/.privé/cvs$ export CVSROOT=:ext:nroche@narval.hd.free.fr:/cvsroot
nroche@planet-gis:~/.privé/cvs$ export CVS_RSH=/usr/bin/ssh
nroche@planet-gis:~/.privé/cvs$ export CVSEDITOR=vi
nroche@planet-gis:~/.privé/cvs$ cvs checkout narvali
```

3 Update

cvs update
Permet de récupérer la dernière version.
Exemple :

```
nroche@narval:~/src/narvali$ cvs update
```

3.1 Simulation

L'option `-n` permet de simuler la requête et donc de voir quels fichiers seront ajournés.

```
$ cvs -n update
```

3.2 Nouveau répertoire

Pour intégrer les nouveaux répertoires il faut utiliser l'option `-d` ou sinon :

```
nroche@narval:~/src/narvali$ cd ..
nroche@narval:~/src/narvali$ cvs checkout narvali/latex
```

4 Diff

Permet de voir les fichiers ont été modifié depuis le dernier commit.

```
$ cvs diff
```

5 Commit

cvs commit [FILE]

5.1 Ajout

```
$ touch toto
$ cvs add toto
$ cvs commit toto
```

5.2 Retrait

```
$ rm toto
$ cvs remove toto
$ cvs commit toto
```

6 Merge

```
$ cp toto toto_a_moi
$ cvs update toto
 <merge>
$ cvs commit toto
```

7 Déplacer un répertoire

On est obligé d'intervenir côté serveur, où un simple MV suffit. Côté client, si l'on ne souhaite pas faire de **checkout** (commit non fait), alors on doit en plus du déplacement du répertoire adapter les fichiers CVS.

- Déplacement côté serveur.

  ```
  nroche@narval:/cvsroot/narvali$ mv apache2/ admin/
  ```

- Déplacement côté client, si l'on ne veut pas faire de **checkout**.

  ```
  nroche@planet-gis:~/.privé/cvs/narvali$ mv apache2/ admin/
  ```

- Déplacer la ligne suivante du fichier *CVS/Entries* au fichier *CVS/admin/Entries*

  ```
  D/apache2////
  ```

- Adapter la ligne du fichier *CVS/admin/apache2/Repository*

  ```
  narvali/admin/apache2
  ```

8 Récupération des sources

8.1 Logs

```
$ cvs log gigabit/page.tex
head: 1.6
...
----------------------------
revision 1.6
- maj gigabit
- ajout du serveur de console
----------------------------
revision 1.5
```

```
- maj gigabit
- ajout du serveur de console
----------------------------
revision 1.4
test suite au mail envoyé à Rupert
----------------------------
revision 1.3
serveur DHCP + mise à jour de l'article global (book).
----------------------------
revision 1.2
blabla cpci
----------------------------
revision 1.1
Premiers tests des chassis PCI depuis l'environnement Linux
```

8.2 Diff

```
$ cvs diff -r 1.5 page.tex
retrieving revision 1.5
retrieving revision 1.6
diff -r1.5 -r1.6
...
13c13
<   {\pageTitle{Gigabit}}
---
>   {\pageTitle{Serveur de consoles}}
...
```

8.3 Récupération

La méthode donnée ci-dessous consiste à commiter l'ancien contenu et donc à incrémenter le numéro de version. **Attention**, le numéro de version est inscrit en entête de chaque fichier. Il ne faut pas écraser les fichiers car on doit impérativement récupérer le dernier numéro de version.

```
$ cd ..
$ cvs update -r 1.5 gigabit

$ less gigabit/page.tex                  # vérif
$ cp gigabit/page.tex /tmp/page.tex

$ cvs update -r 1.6 gigabit
$ cat /tmp/page.tex >> gigabit/page.tex
$ vi /tmp/page.tex gigabit/page.tex      # laisser le bon entête !!
$ cvs commit gigabit
```

9 Astuces

9.1 Via Emacs

```
# apt-get install ssh-askpass
```

M-x cvs-examine puis C-h m ou plutôt C-h b

9.2 Groupe

Afin de rendre possible la mise à jour concurente depuis et sur le serveur, il faut affecter aux utilisateurs le même groupe dans le fichier */etc/password*.

9.3 Groupe pour la copie de travail

- dépôt

```
$ cd tmp
$ install -d cvsroot
$ cvs -d $PWD/cvsroot init

$ mkdir src
$ cd src
src$ touch test.txt
src$ cvs -d $PWD/../cvsroot import -m "comment" module moi v1
src$ cd ..
```

- comportement normal : pas de group en écriture

```
$ install -d cvs
$ cd cvs
cvs$ cvs -d $PWD/../cvsroot co module
cvs$ ls -l module/test.txt
-rw-r--r-- 1 nroche nroche 0 févr.  8 17:22 module/test.txt
```

- umask

```
$ install -d cvs
$ cd cvs
cvs1$ umask 007
cvs1$ cvs -d $PWD/../cvsroot co module
cvs1$ ls -l module/test.txt
-rw-rw-r-- 1 nroche nroche 0 févr.  8 17:22 module/test.txt
```

- setgid

```
# addgroup rd
# install -o nroche -g rd -m 02770 -d cvs
$ cd cvs
cvs1$ umask 007
cvs1$ cvs -d $PWD/../cvsroot co module
cvs1$ ls -l module/test.txt
-rw-rw-r-- 1 nroche rd 0 févr.  8 17:22 module/test.txt
```

- update

```
# install -o nroche -g rd -m 02770 -d cvs2
$ cd cvs2
cvs2$ umask 007
cvs2$ cvs -d $PWD/../cvsroot co module
cvs2$ cd module
cvs2/module$ echo "test1" > test.txt
cvs2/module$ cvs commit -m "test"
```

```
cvs1/module$ cvs update
U test.txt
cvs1/module$ ls -l
-rw-rw-r-- 1 nroche rd 27920 févr.  8 18:58 test.txt

# install -o nroche -g rd -m 02770 -d cvs3
$ cd cvs3
cvs3$ umask 007
cvs3$ cvs -d :ext:nroche@localhost:$PWD/../cvsroot co module
cvs3/module$ cvs update
P test.txt
cvs3/module$ ls -l
-rw-r----- 1 nroche rd 27920 févr.  8 18:57 test.txt
```

remarque : L'envoit d'un patch (P vs U) dans le cas d'une connexion ssh ne prend pas en compte le umask.

9.4 Développement avec 2 archives CVS

Attention, faire pointer vos espaces de développement sur l'autre serveur ne fonctionne pas. Il faut recopier les fichier depuis et vers les espaces de développement puis 'comiter'.
Au delà d'un certain seuil de différence, il faut recopier le module */cvsroot/xxx* d'un serveur sur l'autre.

9.5 Entête CVS

Ajouter `$Id$` : dans les fichiers. Par exemple en et suite aux commits :

```
% $Id: page.tex,v 1.14 2014-04-08 16:37:41 nroche Exp $
```

10 Access anonyme

ici ou la

```
# apt-get install xinetd
# /etc/init.d/xinetd stop
# cat >> /etc/xinetd.d/cvs << EOF
service cvspserver
{
    port = 2401
    socket_type = stream
    protocol = tcp
    user = root
    wait = no
    type = UNLISTED
    server = /usr/bin/cvs
    env = HOME=/var/lib/mdtx/cvsroot
    server_args = -f --allow-root /var/lib/mdtx/cvsroot pserver
    disable = no
}
EOF

# /etc/init.d/xinetd start
```

```
$ telnet localhost 2401
# useradd -s /bin/false anonymous
# useradd anonymous -s /bin/false -u 98
$ echo "anonymous:" >> /srv/cvsroot/CVSROOT/passwd
$ echo "anonymous" >> /srv/cvsroot/CVSROOT/readers
$ chmod o+rwx /var/lib/mdtx/cvsroot/salut
$ cvs -d :pserver:anonymous@localhost:/var/lib/mdtx/cvsroot login
```