

Ceci est un petit récapitulatif du cours.

Bientôt deux autres documents regroupant toutes les commandes utiles CAD celle-ci plus quelques unes non utilisées au cours mais fort bien sympathiques arriveront. Arrivera également un tutorial fort complet de VI. Enfin ceci n'est pas le travail fini. Le sur lignage jaune indique que c incomplet. Sachez encore que plus d'exemples viendront agrémenter les syntaxes. Mais bon pour que vous révisiez, voici toujours ceci.

Les systèmes de fichier.

Les fichiers de périphériques.

NB : les lecteurs sont des répertoire se trouvant dans le répertoire /dev situé sur la racine.

Disque dur :

Hdd master primaire : hda

Hdd slave primaire : hdb

Hdd master secondaire : hdc

Hdd slave secondaire : hdd

Partitions primaires numérotées de 0 → 4

Lecteur logique dans la partition étendue de 5 → ...

Ex : hda1, hda2, hdc6

Lecteur de disquette :

fd0, fd1

Précision de la capacité (si nécessaire) : fd0h720, fd0h1440

Lecteur CDROM, DVD, graveurs, ... :

Comme les disques durs (sans la numérotation derrière le périphérique). OU ALORS cdrom

Ex : hdd, cdrom

Imprimante : lptx où x est un numéro

Carte réseau : ethx où x est un numéro

Interface sériele tysx où x est un numéro

Les principes des liens.

Lien dur	Lien symbolique
Indique la position du fichier	Equivaut au raccourci sous windows
Même inode	Inode différent
Syntaxe : ln source cible	Syntaxe : ln -s source cible
Ex : ln origine.txt liendur.txt	Ex : ln -s origine.txt liensymbolique.txt

Les points de montages.

Monter un périphérique :

Syntaxe la plus couramment utilisée :

mount -t type device répertoire

Type : vfat → fat32, msdos → fat16, iso9660 → cdrom, etc

```
Ex : mount -t msdos /dev/fd0h1440 /home/floppy
      mount -t vfat /dev/hdb1 /home/windows
      mount -t iso9660 /dev/cdrom /home/cdrom
```

Démonter un périphérique :

Syntaxe : umount répertoire

Ex : umount /home/floppy

Création d'un système de fichier.

Formatage de bas niveau :

fdformat

Permet le formatage bas niveau d'une disquette

Syntaxe : fdformat [-n] périphérique

1 seule option → -n aucune vérification après sauvegarde.

Système de fichier :

Il est important de bien choisir un système de fichier. Ceci en fonction des métadonnées et surtout s'il faut ou non un journal. Ceci est sujet à beaucoup de discussions aussi vous cherchez vos propres informations pour choisir celui qui vous conviendra le mieux.

Sachez juste que msdos équivaut à la fat16, vfat à la fat32. En Unix on retrouve l'Ext2, l'Ext3, le ReiserFS, le JFS et l'XFS. Le ReiserFS et l'Xfs ont de bonnes performances. Si vous avez de petits fichiers choisissez le premier.

Commandes : mkfs, mkdosfs, mke2fs, mkfs.vfat, mkfs.xfs

Manipulation des fichiers.

Man

Permet d'avoir toutes les informations relatives à une commande.

Syntaxe : man [commande]

Ex : man ls

Ls

Permet de lister le contenu d'un ou plusieurs répertoires.

Syntaxe : ls [option]... file...

Option : -a affiche également les fichiers commençant par un .(point)

-i affiche l'inode

-R récursivité effectuée

-l affiche de nombreuses informations diverses :

Ex : drwxr-xr-x 2 root root 48 Sep 21 19:37 cdrom

Nous retrouverons dans l'ordre de gauche à droite:

- Type du lien ('-' pour un fichier 'd' pour un répertoire 'l' pour un

lien symbolique)

- Autorisations d'accès (aspect analysé lors d'un prochain chapitre)
- compteur de lien (voir paragraphe 1.2 traitant des liens)
- propriétaire du fichier
- groupe auquel le fichier appartient
- taille du fichier
- date de dernière modification du fichier.
- nom du fichier ou répertoire.

A noter l'inode s'affiche avant tout ceci.

Ex : `ls -il`
`ls -lR`

CP

Permet de copier un fichier.

Syntaxe : `cp [option]... source destination` (1)
`cp [option]... source... répertoire` (2)

1 : copie dans un même répertoire en changeant le nom

2 : copie de plusieurs fichiers dans un autre répertoire

Option : `-r` copie récursive, permet la copie de répertoire et sous répertoire
`-l` effectue un lien dur
`-s` effectue un lien symbolique
`-i` demande une confirmation lors de la copie
`-f` effacer les fichiers existants

Ex : `cp fichier1.txt copie/text1.txt`
`cp fichier1.txt fichier2.txt copie`
`cp *.* copie`
`cp -s ../*.*/ ./`

Rm

Permet de supprimer un ou plusieurs fichiers.

Syntaxe : `rm [option]... FILE...`

Option : `-r` suppression récursive
`-i` confirmation demandée lors de la suppression
`-f` pas de confirmation, pas de message de diagnostique et pas d'erreur si le fichiers n'existe pas

Ex : `rm test.txt`
`rm -i ../.*`
`rm -r répertoire`

Mv

Permet de déplacer ou renommer un fichier

Syntaxe : mv [option]... source destination (1)
 mv [option]... source... destination (2)

1 : renommer
2 : déplacer

Option : -i demander confirmation
 -f écraser fichier de destination

Ex : mv test.txt renomme.txt
 mv text1.txt test2.txt répertoire/
 mv test.txt répertoire/renomme.txt

Manipulation des répertoires.

Mkdir

Créer un ou plusieurs répertoires.

Syntaxe : mkdir [option]... répertoire...

Ex : mkdir répertoire1

Rmdir

Permet de supprimer un répertoire SI celui-ci est VIDE

Syntaxe : rmdir [option]... répertoire...

Option : -p efface les répertoires parents tant que ceux-ci sont vides

Ex : rmdir répertoire1
 rmdir -p répertoire1

Cd

Permet de changer de répertoire

Ex : cd test
 cd ../../test/soustest
 cd ./test

Pwd

Permet de connaître le répertoire courant

Syntaxe : `pwd`

Find

Permet la recherche de fichier. Peut s'utiliser avec le filtre grep.

Syntaxe : `fin [chemin] [expression]...`

Expression : composée d'options et/ou tests et/ou actions.

La commande find permet tant de choses et peut avoir tant de différentes options intéressantes que je vous renvoi au manpages.

Ex : `find ./ -xtype l -exec rm {} \;` (supprime les liens symboliques ne pointant sur rien)
`find / -type f -name *.conf -exec grep -H nights {} \;`
`find ./ -type f -user student`
`find ./ -type f -size +400k`

Filtres

Permet l'exécution d'une routine dans une autre.

Syntaxe : `action | action`

Ex : `find ./ -type f | wc -l`

La structure de fichier.

<code>/home</code>	contient le répertoire de travail (fichiers, paramètres,...)
<code>/root</code>	répertoire de l'administrateur
<code>/usr</code>	contient la majeure partie de l'os
<code>/bin</code> <code>/sbin</code>	programmes liés à l'os nécessitant d'être dans l'amorçage (peut se trouver dans <code>/usr</code>)
<code>/etc</code>	fichiers de configurations système (réseau, utilisateur, équivalent des batch,...)
<code>/boot</code>	contient le kernel (peut s'appeler aussi <code>/stand</code> ou <code>/unix</code>)
<code>/lib</code>	librairies dynamiques (peut se trouver dans <code>/usr</code>)
<code>/var</code>	élément dont le contenu change régulièrement tels que par ex : <ul style="list-style-type: none">• journaux d'évènements <code>/var/log</code>• queue d'impression <code>/var/spool</code>

- configuration DNS /var/dns

Les droits d'accès aux fichiers et répertoires.

Ex : 23564 drwxr-xr-x 2 root root 48 Sep 21 19:37 cdrom

Nous retrouverons dans l'ordre de gauche à droite:

- L'inode
- Type du lien ('-' pour un fichier 'd' pour un répertoire 'l' pour lien symbolique)
- Autorisations d'accès (9 caractères)
- compteur de lien
- propriétaire du fichier
- groupe auquel le fichier appartient
- taille du fichier
- date de dernière modification du fichier.
- nom du fichier ou répertoire.

Autorisation d'accès :

Les trois premiers caractères concernent le propriétaire du fichier ou répertoire.

Les trois caractères suivants concernent le groupe propriétaire du fichier ou répertoire.

Les trois derniers caractères concernent « le reste du monde ».

	Fichiers	Répertoires
r	Lisible : copier, coller, imprimer	Contenu visible
w	Editer, modifier, renommer, supprimer	Placer des fichiers et des sous répertoires
x	Exécuter	Traversable : on peut accéder à des fichiers et des sous répertoire même si on ne voit pas ce que contient le répertoire

Editeur Vi.

Vi est un puissant éditeur de texte. Même si celui-ci paraît repoussant ou compliqué, il vaut vraiment la peine qu'on s'y attarde. Sachez qu'en maîtrisant entièrement VI vous pouvez absolument tout faire ce qui est possible avec un éditeur de texte. Toutefois n'ayant le temps ou même l'utilité de maîtriser entièrement VI, voici quelques commandes principales permettant une utilisation plus aisée de l'éditeur de texte VI.

Syntaxe de lancement : vi nom_du_fichier
 vim nom_du_fichier

A noter : si le fichier ne se trouve pas dans le répertoire courant il faut indiquer le chemin complet du fichier, sinon celui-ci sera créé.

Se déplacer dans VI : - les flèches multi – directionnelles
 - h,j,k,l (utile pour connexion telnet par ex.)

Les deux modes...

Vi fonctionne sous deux modes : le mode normal et le mode insertion. Le mode normale permet de taper des commandes propres à VI. Le mode insertion permet quant à lui de modifier le texte.

Pour commuter du mode insertion au mode normale pressez la touche Esc.

Pour commuter du mode normale au mode insertion :

- Insert ou I => passe en insertion à l'endroit où se trouve le curseur.
- A => positionne le curseur à la fin de la ligne courante

Quelques commandes de base

Quitter sans sauver → :q !

Enregistrer → :w ou :w ! (*permet d'écraser le fichier*)

Enregistrer et quitter → :wq

Annulation (undo) → :u

Effacer le caractère derrière le curseur → x

Effacer la ligne où se trouve le curseur → dd

Pour plus d'info sur VI voir document annexe : VI.pdf