

Fichiers



Laurent Réveillère

Enseirb-Matmeca
Département Télécommunications

`Laurent.Reveillere@ipb.fr`
`http://uuu.enseirb-matmeca.fr/~reveille/`



Fichier

Ensemble d'informations regroupées en vue de leur conservation et de leur utilisation dans un système informatique

□ Description

- Suite d'octets
- Nom
- Ensemble de fonctions d'accès pour les manipuler



Fichiers sous UNIX

- Tout dans Unix est fichier
 - tout objet permettant de manipuler un flot d'octets

- Types de fichiers sous UNIX
 - Fichiers ordinaires
 - Répertoires
 - Fichiers spéciaux (périphériques, tubes, liens, sockets)



Arborescence et chemins

- Répertoires
 - Contient un ensemble de répertoires et de fichiers
 - Possède un et un seul père
- Répertoire racine
 - Seul répertoire n'ayant pas de père
- Chemin
 - Unique pour chacun des descendants
 - Concaténation de tous les répertoires en partant de la racine



Chemins absolus et relatifs

□ Chemin relatif

➤ Exprimé depuis le répertoire courant

- » .. Désigne le répertoire père
- » . Désigne le répertoire courant

➤ Exemple

Foo/bar

./Foo/bar

./Foo/../Foo/bar

□ Chemin absolu

➤ Exprimé de puis la racine → unique

➤ Exemple

/net/exemple/Foo/bar



Systeme de Gestion des Fichiers (SGF)

- ❑ Réalise la correspondance entre
 - organisation logique (vue utilisateur) et
 - organisation physique
- ❑ Optimise l'utilisation des disques
 - Espace
 - Temps
- ❑ Assure le stockage permanent des données
 - Intégrité : blocs endommagés, corruption, etc
 - partage et protection (attributs, droits)



Terminologie

- Disque
 - Support non volatile de données
 - Unité d'allocation : secteur
 - organisé en secteurs, têtes et cylindres
- Partition
 - Portion du disque
- Volume
 - Une ou plusieurs partitions
 - Comporte un système de fichiers
- Point de montage
 - Point désiré du répertoire général
 - Intégration du répertoire d'un volume

Shéma

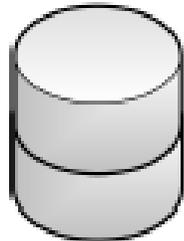
Disque 1



/home

/var

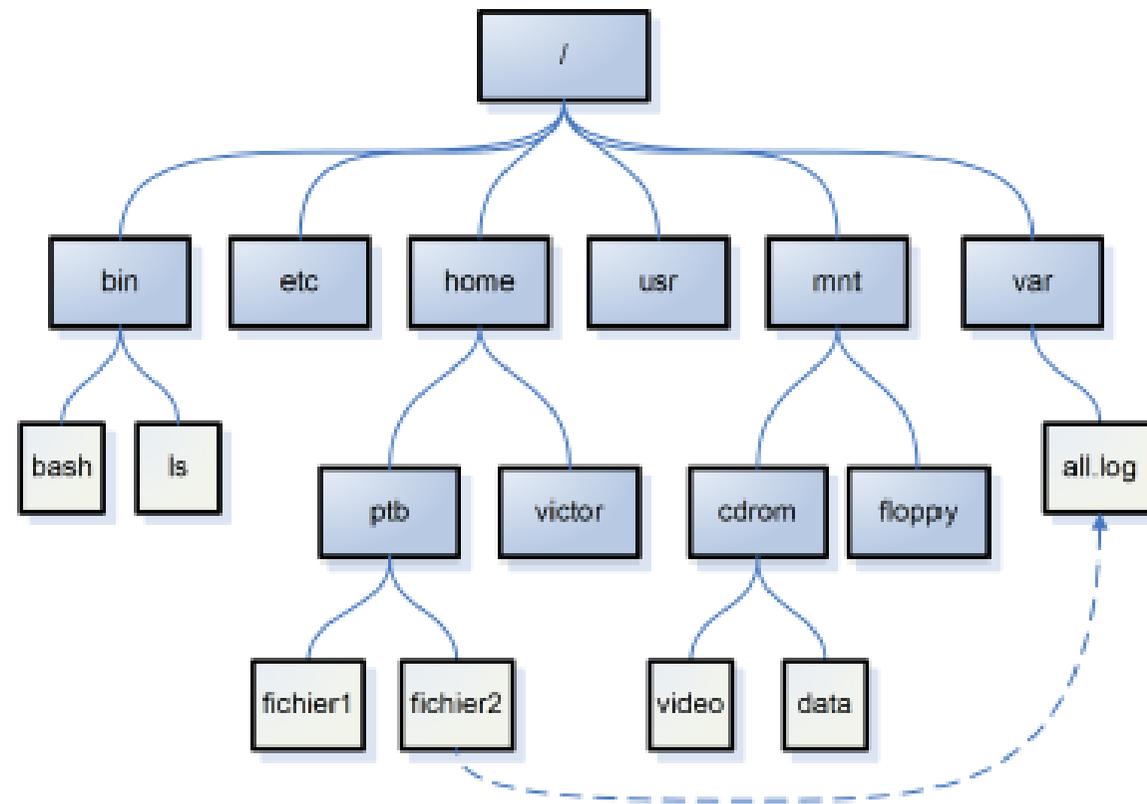
Disque 2



/usr

/

CD-Rom





Outils pour SGF

□ mkfs

- Permet d'installer un système de fichiers dans un volume

```
mkfs -t ext2 /dev/sda1
```

□ Mount

- Permet de monter un volume dans un point de montage
- Point de montage = répertoire de l'arborescence

```
mount -t vfat /dev/sda1 /mnt/Mes_Documents
```

- /etc/fstab
 - Fichier de configuration du système
 - Montage automatique des volumes à chaque démarrage

- Commande df
 - Affiche la liste des volumes montés
 - Exemple d'utilisation

```
df -T -h
```



Systeme de fichiers

- Méthode d'organisation des fichiers et répertoires sur un volume
 - Découpe le volume en blocs (ou cluster) qui est l'unité d'adressage
 - Comporte un certains nombres de blocs réservés au système pour sa propre gestion (espace libre...)
 - ➔ Tous les autres blocs sont des blocs de données
 - Mets en œuvre des algorithmes ad-hoc pour organiser les données

- Linux
 - Ext2, ext3, ext4 ...



Algorithmes de SGF

□ Différentes stratégies

- Allocation contigüe
- Allocation par liste chaînée
- Allocation par liste chaînée avec table d'allocation
- Allocation par noeud d'index (index-node ou inode)

□ Conséquence

- Fragmentation



Fragmentation

- ❑ Effet de bord des algorithmes de stockage
- ❑ Rend indisponible une partie de l'espace sur le volume
- ❑ Applicable à tous les types de SGF
 - Taux de fragmentation arrive à plus ou moins long terme
 - Dépend du profil d'usage et de l'aglotithme



Type de fragmentation

- Fragmentation interne
 - Utilisation d'une taille de bloc fixe
 - Espace alloué supérieur à l'espace nécessaire
 - Espace dans certains blocs non utilisable

- Fragmentation externe
 - Somme des blocs disponibles suffisant
 - Blocs non contigus



Inode

- ❑ Représentation interne d'un fichier
- ❑ Spécifique à un volume
- ❑ Ne contient pas le nom du fichier
 - Droits d'accès et type du fichier
 - Numéro du propriétaire, groupe
 - Taille du fichier
 - Date et heure du dernier accès en lecture et écriture
 - Date et heure de dernière modification de l'inode
 - Nombre de liens physiques
 - Pointeurs vers les blocs de données
- ❑ Voir la commande `stat`



Gros plan sur un inode

- ❑ Chaque inode contient une série de pointeur
 - Les premiers pointent vers des blocs de données du fichier
 - Les derniers, sont des indirections : ils pointent vers des blocs d'index qui contiennent une nouvelle série de pointeurs directs (données) ou indirects (blocs d'index)
- ❑ Suppression d'un fichier
 - Chaque Inode contient un compteur de références
 - Suppression effective lorsque compteur de ref. égale à 0
 - Suppression de l'inode, pas des données
 - » Données toujours disponibles si aucune écriture dessus



Fichier répertoire

- ❑ Fichier contenant la liste des fichiers (ou répertoires) qu'il contient
 - Inode
 - Nom du fichier
- ❑ Restriction sur la taille des noms de fichiers
 - < 256 caractères



Fichiers et liens

□ Lien physique

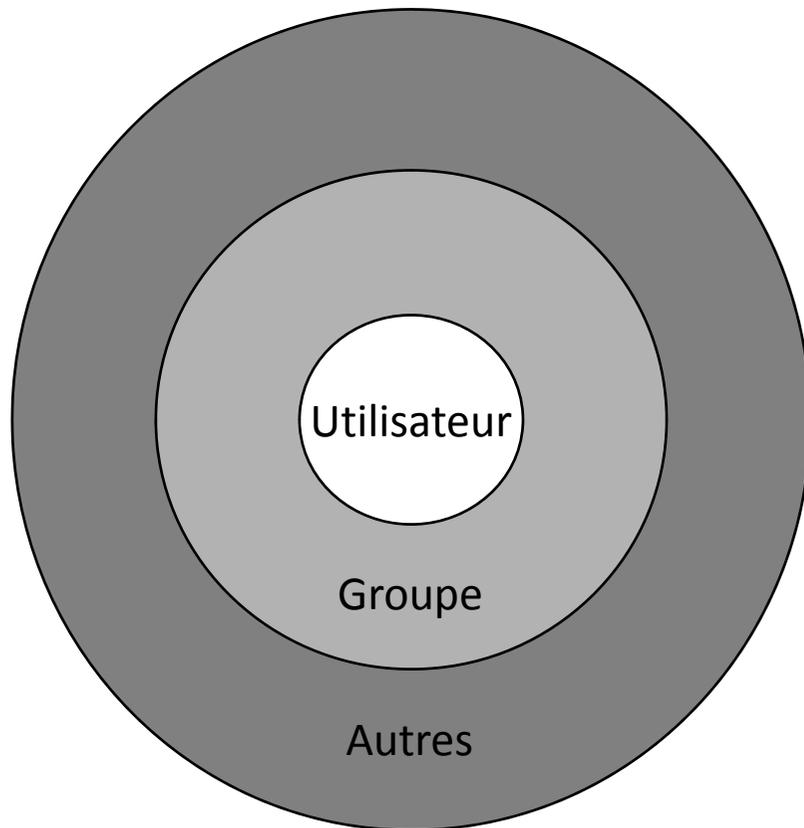
- Association nom fichier et inode
- Inode peut être référencé plusieurs fois
- Numéros d'inodes spécifique à un volume
 - » Lien physique utilisable quand dans un même volume

□ Lien symbolique

- Fichier spécial contenant le chemin du fichier référencé
- N'incrémente pas le compteur de référence d'un inode
- Cas d'un fichier effacé
 - » Lien pointant dans le vide, *dangling*



Autorisation : abstraction fichier



- ❑ Abstraction fichier
 - Simple
 - Répertoire
 - Périphérique
- ❑ Niveaux
 - Utilisateur
 - Groupe
 - Autres
- ❑ Protections
 - Lecture
 - Ecriture
 - Exécution



Autorisation : fichier simple

- Lecture

- permission de consulter le contenu

- Ecriture

- permission de modifier le contenu

- Exécution

- permission d'exécuter le fichier en tant que programme



Autorisation: répertoire

Lecture

- permission de consulter le contenu d'un répertoire

Ecriture

- permission d'ajouter ou de supprimer des fichiers dans le répertoire

Exécution

- permission de se déplacer dans ce répertoire



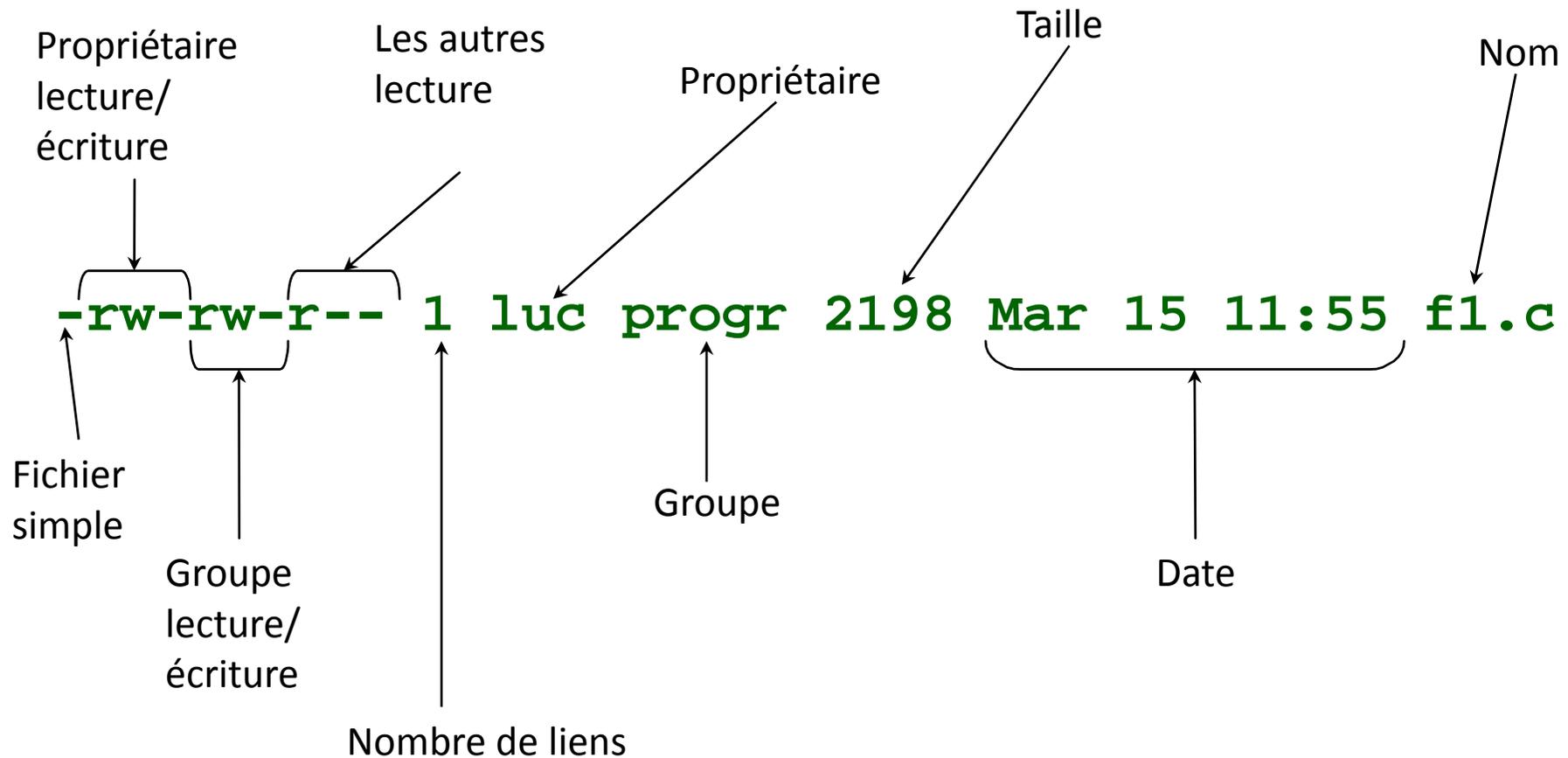
Autorisation: les niveaux

- Utilisateur
 - celui qui crée le fichier (généralement)
- Groupe
 - un ensemble d'utilisateurs (par ex. une équipe de programmeurs)
- Autres
 - les utilisateurs qui ne sont ni l'utilisateur ni membres du groupe



Exemple : fichier simple

```
ls -l
```





Exemple : executable

Propriétaire
lecture/
écriture/
exécution

Les autres
lecture/
exécution

```
-rwxr-xr-x 1 luc progr 316 Mar 15 14:55 truc
```

Fichier simple

Groupe lecture/
exécution



Exemple : répertoire

Propriétaire
lecture/
écriture/
exécution

Les autres
lecture/
exécution

```
drwxr-xr-x 3 luc progr 316 Mar 15 14:55 proj
```

Répertoire

Groupe
lecture/
exécution



Changement des droits d'accès

□ **chmod MODE fichier**

➤ **MODE**

» ugoa

» +/-

» rwx

□ **chown propriétaire fichier**

□ **chgrp groupe fichier**

(Exemple: installation d'un programme dans **bin** en tant que superutilisateur)

Shell : personnalisation de l'environnement



Laurent Réveillère

Enseirb-Matmeca
Département Télécommunications

`Laurent.Reveillere@ipb.fr`
`http://uuu.enseirb-matmeca.fr/~reveille/`



Qu'est ce qu'un environnement ?

Environnement

Ensemble de concepts qui expriment ce que fait la machine en termes compréhensibles et cohérents, avec une apparence agréable et une ergonomie confortable.

Personnalisation de l'environnement

Rendre un environnement spécifique/adapté à une personne donnée



Qu'est qu'un environnement Shell Unix

- ❑ Fichiers
- ❑ Répertoires
- ❑ Entrée/sortie standard
- ❑ Commandes Unix : protection, impression, etc.
- ❑ Ergonomie Unix :
 - Organisation des répertoires
 - Emplacement de chaque type de fichier
 - Nom des fichiers, répertoires et commandes



Personnalisation de Bash

- Les fichiers spéciaux (fichiers de démarrage)
- Les alias (synonymes de commandes)
- Les options (aspects de l'environnement)
- Les variables (valeur des paramètres de fonctionnement du Shell)



Fichiers spéciaux

.bash_profile .bash_logout .bashrc

- Emplacement : dans votre répertoire
- Fonction :
 - Configuration automatique
 - » En début de session
 - » Au lancement d'un nouveau Shell
 - Invocation de commandes à la fin d'une session
- Par défaut : **/etc/.profile**



Fichier spécial : `.bash_profile`

- ❑ Lu à l'entrée sur le système
- ❑ Exécution des commandes qu'il contient
- ❑ Exemple typique

```
PATH=/sbin:/usr/sbin:/bin:/usr/bin:/usr/local/bin
```

```
SHELL=/bin/bash
```

```
MANPATH=/usr/man:/usr/X11/man
```

```
EDITOR=/usr/bin/vi
```

```
PS1='\h:\w$ '
```

```
PS2='> '
```

```
export EDITOR
```

- ❑ Prise en compte des modifications:

```
source .bash_profile
```



Fichier spécial : `.bashrc`

- Lu et exécuté lors du lancement d'un autre Shell (**bash**)
- Avoir des configurations compatibles
- dans `.bash_profile` mettre
`source .bashrc`
- Par défaut : rien ne s'exécute



Fichier spécial : `.bash_logout`

- ❑ Lu et exécuté chaque fois qu'une session se termine (login Shell)
- ❑ Exemples d'utilisation : supprimer les fichiers inutiles, consigner la durée de la session, ...
- ❑ Par défaut : rien n'est exécuté



Les alias

- ❑ Renommage de commandes

- ❑ Syntaxe

alias NOM=COMMANDE

- ❑ Définition

- ligne de commande

- **.bash_profile**

- **.bashrc**

- ❑ Exemple

- **alias liste=ls**

- **liste**

fich1 fich2 donnees



Les alias

□ Exemple

➤ `alias cdproj1='cd ~/projets/proj1'`

➤ `alias ll='ls -l'`

□ Syntaxe :

➤ Pas d'espace

➤ Protection

□ Fonctionnement

➤ Recherche du mot au début d'une chaîne de commande

➤ Remplacement du mot par l'alias

➤ Expansion éventuelle des caractères génériques



Les alias

❑ Exemple avec caractères génériques

➤ `alias printall='pr * | lpr'`

❑ Exemple récursif

➤ `alias pa=printall`

❑ Exemples invalides

➤ `alias ls='ls -l'`

➤ `alias liste`

`alias ls=listfile`



Les alias : commandes

- Affichage d'un alias

- **alias lf**

```
alias lf='ls -F'
```

- Affichage de tous les alias

- **alias**

```
alias lf='ls -F'
```

```
alias printall='pr * | lpr'
```

```
...
```

- Suppression d'un alias

- **unalias lf**



Les options

- ❑ Comportement du Shell
- ❑ Deux états : **on** (-o) / **off** (+o)
`set -o OPTION`
`set +o OPTION`
- ❑ Affichage de toutes les options
`set -o`

Options	Description
emacs	Mode Emacs pour l'édition de ligne
vi	Mode vi pour l'édition de ligne
ignoreeof	Interdit CTRL-D pour fin de session ; utilisation de exit
noglob	Non-expansion des caractères génériques
nounset	Erreur lors de l'utilisation d'une variable non-définie
noclobber	Interdit l'écrasement de fichier lors d'une redirection



Les variables

- ❑ Variables : sauvegarder des valeurs
- ❑ Convention
 - Nom en lettres majuscules :
 - » Variables spécifiques à Bash
 - » Variables d'environnement (voir plus tard)
 - Nom en lettres minuscules : les autres
- ❑ Syntaxe définition
VARIABLE=VALEUR
- ❑ Syntaxe utilisation
\$VARIABLE
- ❑ Syntaxe suppression (voir option **nounset**)
unset VARIABLE



Les variables : exemples

- ❑ Définition d'une variable
 - `verre=eau`
- ❑ Affichage de la valeur d'une variable
 - `echo $verre`
`eau`
- ❑ Variable et protection (faible/forte)
 - `echo "la valeur de \ $verre est \ "$verre\"."`
`la valeur de $verre est "eau".`
 - `phrase="la valeur de \ $verre est \ "$verre\"."`
 - `echo $phrase`
`la valeur de $verre est "eau".`
 - `phrase='la valeur de \ $verre est \ "$verre\".'`
 - `echo $phrase`
`la valeur de \ $verre est \ "$verre\".`



Exemples de variables internes

- ❑ Variables pour l'édition de ligne
 - **HISTCMD** : numéro de la commande courante dans l'historique
 - **HISTFILE** : nom du fichier d'historique
- ❑ Variables pour le courrier électronique
 - **MAIL** : nom du fichier à tester
`/var/spool/mail/votre_nom`
 - **MAILCHECK** : intervalle, en secondes, entre deux tests



Variables de l'invite Shell

- ❑ Les variables de l'invite du Shell :
PS1, PS2 (continuation), **PS3, PS4**
- ❑ Exemples de configuration de l'invite

- Nom d'utilisateur

```
$ PS1="\u>"
```

```
alice>
```

- Numéro d'historique

```
alice> PS1="\u \!>"
```

```
alice 13>
```

- Emplacement

```
alice 15> PS1="\u \! \w>"
```

```
alice 16 ~/projet>
```



Variable de recherche de commandes

- ❑ Commande : un fichier de code executable
- ❑ Emplacements de commandes :
 - Standards
 - `/bin` `/usr/bin` `/usr/local/bin`
 - Autres
 - `/distraction/jeux`
 - `/home/alice/bin`
- ❑ La variable **PATH**



La variable `PATH`

- ❑ Structure et ordre des chemins de recherche

```
$ echo $PATH
```

```
/bin:/usr/bin:/usr/local/bin
```

- ❑ Ajout d'un chemin

```
$ PATH=$PATH":/home/alice/bin"
```

```
$ echo $PATH
```

```
/bin:/usr/bin:/usr/local/bin:/home/alice/bin
```

- ❑ Identification d'un chemin

```
$ type ls
```

```
ls is /usr/bin/ls
```



Variables diverses

- HOME** : nom de votre répertoire personnel
- BASH** : chemin d'accès à cette instance du Shell
- PWD** : répertoire courant



Personnalisation et sous-processus

Lancement d'une commande (ou d'un Shell)

- Exécution d'une commande interne : appel de procédure (ex. `cd`), ou
- Exécution d'une commande externe : création d'un sous-processus
 - » Propagation de tout ou partie de l'environnement
 - Variables
 - Options
 - Alias



Variables d'environnement

- ❑ Variables internes : **BASH, HOME, MAIL, PATH, PWD, TERM, EDITOR, ...**
- ❑ Informations importantes pour sous-processus : éditeur de texte, gestionnaire de courrier électronique, ...
- ❑ Création de variables d'environnement
export VAR1 [VAR2 ...]
- ❑ Affichage des variables d'environnement (et leur valeur)
export



Les fichiers `.bashrc` et `.bash_profile`

□ Fichier `.bashrc`

- Les options
- Les définitions d'alias

□ Fichier `.bash_profile`

- Définitions de variables d'environnement et leur exportation
- Des commandes diverses (par ex. affichage)



Exemple de fichier `.bash_profile`

```
stty intr ^C
date
from
ENV=~/.mybashrc
export ENV
source .bashrc
```