

Cours UNIX

le système de fichiers

source <http://www.linux-france.org/article/dalox/>

Jean-Claude Bajard

IUT - université Montpellier 2

Concepts de base UNIX

- **noyau** : cœur du système d'exploitation
 - * gestion des périphériques : disques, réseau,...
 - * gestion des programmes : lancement, exécution
 - * allocation de la mémoire, du cpu...
- **programmes système**
 - * différents services d'un système d'exploitation
 - * utilisent les outils fournis par le noyau (appels système, man syscalls)
- un **interpréteur de commande SHELL** : sh, ksh, bash, csh, tcsh...

Installation d'un système d'exploitation

- faire la liste de la configuration matérielle : moniteur, souris, disques, lecteurs...
- taille des différents systèmes à installer
- organisation des disques : partitions...
- configuration du BIOS pour l'amorce de boot
- configuration du Master Boot Record

partitionnement du disque

Exemple avec un disque de 6Go et 128Mo de ram

- 256Mo pour la racine / (espace relativement fixe)
- 3Go pour /usr (reçoit toutes les applications)
- 512Mo pour /var (beaucoup de fichiers temporaires, mails, impressions...)
- 256Mo pour le swap (doit être capable de faire une copie de la mémoire)
- le reste pour /home (pour les utilisateurs)

Mémoire virtuelle

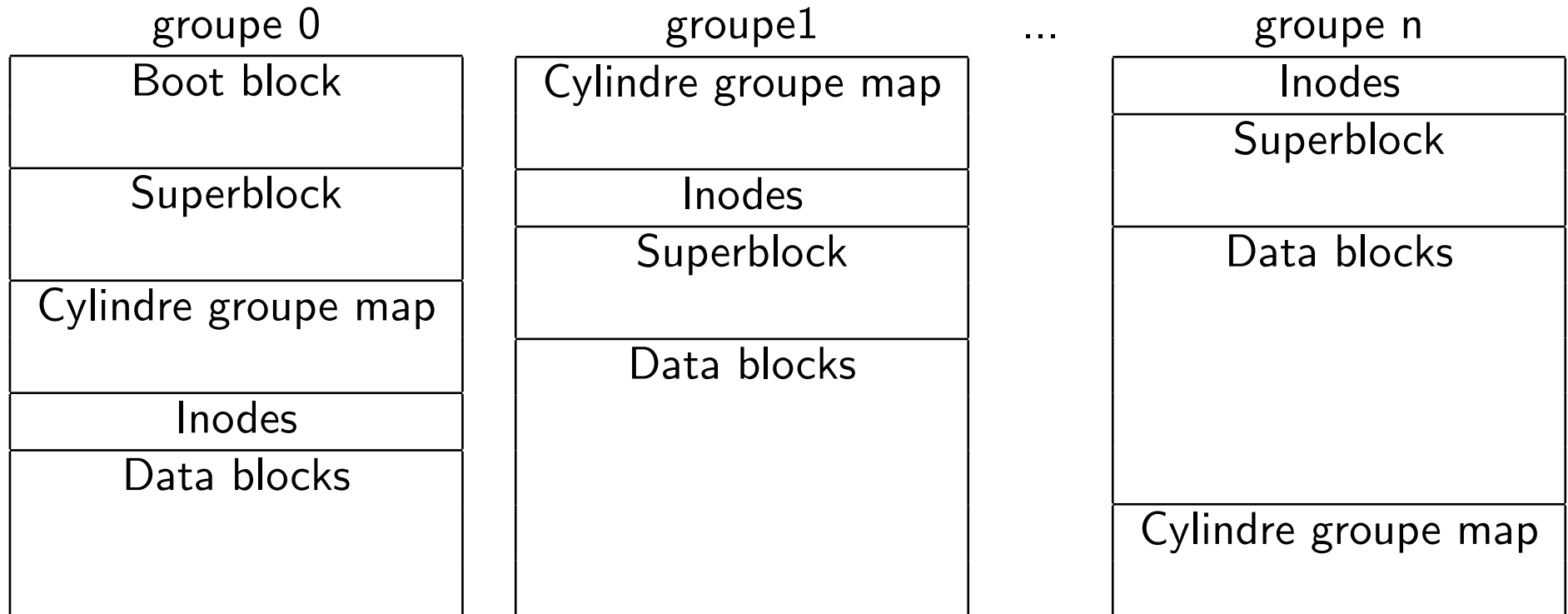
- augmenter l'espace mémoire utilisable par les processus
- découpage en pages, chargement des pages dans la RAM au fur et à mesure des besoins
- Si la RAM est saturée alors copie des pages les plus anciennes dans la zone de SWAP
=> swapper
- la partition SWAP n'a pas de système de fichiers, l'existence d'une zone de swap est obligatoire
- taille en général deux à trois fois la taille de la mémoire. (espace libre avec la commande "free")

Structure physique

Un FS correspond à une partition : `/usr/include/linux/ext2_fs.h`

Boot block	<ul style="list-style-type: none">• Contient le programme de démarrage du système• Sa taille est de 1024 octets.
Superblock	<ul style="list-style-type: none">• informations vitales (état, table inodes, nb de cyl. par grp, etc.)• répliqué dans chaque groupe de cylindre
Inode	Mis à part le nom, contient toutes les infos sur un fichier
Data blocks	Utilisés pour le stockage du contenu des fichiers
Cylindres groupe map	Liste les blocs libres, leurs tailles, et d'autres informations

Structure physique (suite)



Fichier

- toujours désigné par un nom.
- possède un unique inode (certaines informations concernant le fichier).
- possède les fonctionnalités suivantes :
 - * ouverture.
 - * fermeture.
 - * lecture (consultation)
 - * écriture (modification)

Fichier (suite)

Un fichier peut être :

- ordinaire (on utilise parfois le terme "normal") (-)
- un répertoire (d)
- un lien symbolique (l)
- un pseudo-fichier :
 - * accès caractère par caractère (c)
 - * dispositif de communication (p)
 - * accès par bloc (b)

Les inodes

- Type de fichier : -,d, l, c, p, b.
- Droits d'accès : par exemple : rwxr-x—
- UID : effectif du processus créateur ou affecté par chown.
- Taille du fichier.
- atime :date de la dernière lecture.
- mtime :date de la dernière modification.
- ctime :date de la dernière connexion.
- GID : effectif du processus créateur, affecté par chgrp ou hérité du répertoire.
- Nombre de liens (physiques) : c'est à dire nombre de noms.
- Adresse des blocs du fichier....

Adressage des blocs de données

- Direct pour les premières (12) adresses.
- Indirect (pointe sur un bloc de n adresses).
- A deux niveaux d'indirection (donc n^2 adresses).
- A trois niveaux d'indirection (donc n^3 adresses) .

Un bloc de données peut contenir par exemple 256 adresses (bloc de 1024 octets).
soit une taille maxi de 16Go.

Les répertoires

- ce sont des fichiers particuliers
- ils contiennent les noms des fichiers et répertoires
- à chaque nom associent un inode
- l'inode 2 du système de fichier correspond à sa racine

Droits d'accès pour les fichiers

Il y a trois types de propriétaires : l'utilisateur (u), le groupe (g) et les autres (o)

Pour chacun d'eux:

r lecture du contenu

w écriture, modification du contenu

x exécution (script shell, application, programmes compilés...)

Visualisation avec la commande `ls -l`

Droits d'accès pour les répertoires

Il y a trois types de propriétaires : l'utilisateur (u), le groupe (g) et les autres (o)

Pour chacun d'eux:

r lecture du contenu (`ls`)

w créer et supprimer (`rm` , `touch`)

x droit d'accès au répertoire (`cd`)

Droits particuliers

- Set UID

fichier si le fichier est exécutable alors exécution avec droit propriétaire (`passwd`)
rep. Les fichiers (y compris les répertoires) créés dans le répertoire héritent du UID

- Set GID

fichier Set GID si le fichier est exécutable sinon verrouillage obligatoire.
rep. Les fichiers (y compris les rép.) créés héritent du GID du répertoire.

- Sticky Bit

fichier Un programme exécutable sera maintenu en swap après la fin de son exécution.
rep. Un fichier du répertoire ne peut être supprimé que par son propriétaire (`/tmp`).

Les commandes utiles

- pour changer les droits **chmod**
- pour changer le propriétaire utilisateur **chown**
- pour changer le groupe propriétaire **chgrp**
- pour créer un lien (plusieurs noms pour un même fichier) **ln ancien nouveau**
- pour copier **cp an nouv** et pour déplacer (ou changer de nom) **mv anc nouv**
- **ls** pour lister un répertoire, **cd** pour changer de position dans l'arborescence.
- **pwd** pour connaître notre position

L'arborescence UNIX

- Une seule arborescence composée de plusieurs systèmes de fichiers et de périphériques
- **df** et **mount** permettent de visualiser ces montages avec les points de montages
- deux façon pour indiquer un chemin
 - * absolu : en partant de la racine, `cd /usr/lib/ssh`
 - * relatif : en partant du `pwd`, `cd ../../lib` ou `cd ../lib`
 - * `.` répertoire courant `..` répertoire père `~` répertoire "home" (ex "`~toto`")

Hiérarchie du système de fichier

Nous pouvons différencier trois entités:

- Le **répertoire racine /** : système de fichiers root doit être adéquat pour démarrer, reconstituer, rétablir et/ou réparer le système : taille environ 150 Mo.
- **/usr** : toutes les autres applications, doit pouvoir être partagé, en général montable en lecture seulement. taille environ 2Go voire plus
- **/var** : tous les fichiers de données variables. taille environ 500Mo

la racine /

La fonction:

- Pour **démarrer** un système, il doit y avoir les utilitaires de base, la configuration, les informations du chargeur de démarrage, et d'autres données de démarrage essentielles, et aussi pouvoir monter /usr et d'autres parties non-essentiels du système de fichiers.
- Pour **réparer** un système, les utilitaires nécessaires au mainteneur expérimenté pour diagnostiquer et reconstruire un système endommagé doivent être présents sur le système de fichiers root.
- Pour **reconstituer** un système, les utilitaires nécessaires à la reconstitution à partir des sauvegardes système (sur disque, bande, etc.) doivent être présents sur le système de fichiers root.

la racine / (suite)

Le contenu:

- **/bin** : Commandes binaires utilisateur essentielles (pour tous les utilisateurs)
- **/boot** : fichiers statiques du chargeur de lancement
- **/dev** : fichiers de périphériques
- **/etc** : configuration système spécifique à la machine
- **/lib** : bibliothèques partagées essentielles et modules du noyau
- **/mnt** : point de montage pour les systèmes de fichiers montés temporairement

- /proc : système de fichiers virtuel d'information du noyau et des processus
- **/sbin** : binaires système (binaires auparavant mis dans /etc)
- /tmp : Fichiers temporaires

Le répertoire /usr

- /usr/X11R6 : système X Window, Version 11 Release 6
- /usr/X386 : système X Window, Version 11 Release 5, sur les plate-formes x86
- **/usr/bin** : Commandes utilisateurs principales (lpr)
- /usr/dict : listes de mots
- /usr/etc : configuration système à l'échelle d'un site
- **/usr/include** : Répertoire pour les fichiers include standards
- **/usr/lib** : bibliothèques pour la programmation et les packages

- /usr/local : hiérarchie locale
- /usr/man : pages de manuel
- **/usr/sbin** : binaires système standard non essentiels (imapd)
- /usr/share : données indépendantes de l'architecture (gnome)
- /usr/src : code source (kernel-source-2.2.14)

le répertoire /var

- /var/adm : fichiers de rapports et compte-rendus (obsolète)
- /var/catman : pages de manuel formatées localement (optionnel)
- /var/lib : information sur l'état des applications
 - * /var/lib/emacs
 - * /var/lib/games
 - * ...
- /var/local : données variables des logiciels de /usr/local
- /var/lock : fichiers lock

- /var/log : fichiers et répertoires de rapports (dmesg)
- /var/named : fichiers de DNS
- /var/nis : fichiers de la base de données du Service d'Information Réseau (NIS)
- /var/preserve : fichiers sauvés après un crash ou un blocage de ex ou vi
- /var/run : fichiers variables d'exécution
- **/var/spool** : répertoires de spool
 - * /var/spool/lpd
 - * /var/spool/mail
- /var/tmp : fichiers temporaires, utilisés pour garder /tmp petit