

La gestion des mémoires de masse

2015

Sylvain Cherrier

Mémoire de masse

Comment stocker de grandes masses d'informations ?

- Lecture
- Ecriture
- Peu encombrant
- Grande capacité
- Peu d'entretien
- longévité

RAM , papier, pierre, USB, DD,SSD, DVD, BR, bande, etc...
et le Cloud....



Administration des systèmes

2013

2/19

Plusieurs types d'usages

Interactif (*usage 'en direct' sur un PC, un serveur*)

Sauvegarde (*contrainte de temps, de taille, mais accès différents*)

Lecture seule (*diffusion de contenu, sécurité...*)

Distribuée (*cloud, partages...*)

Organisation

- Retrouver l'information dans une telle masse n'est pas chose aisée
- Idée générale : Mise en place d'un dispositif chargé de la gestion des données en échange de contraintes pour l'utilisateur
- Fichier : un nom et un chemin (*path*)

Systeme de Gestion de Fichiers

- Découpe les données en briques de la taille des blocs offerts par le support
- Recense la liste de ces blocs, et en assure la gestion (libre, occupé)
- Utilise deux notions : Fichier (*l'enveloppe de rangement*) et le Répertoire (*Fichier de fichiers*)

SGF : gestion...

- Le SGF d'un support est géré par une partie du Système d'exploitation
- Cette partie (VFS sous Linux) est chargée d'intégrer les supports (= *lire le contenu, comprendre l'organisation, offrir une vue virtuelle de cette organisation, puis gérer l'usage* (ajout de fichier, lecture, modifs...))

Plusieurs FS selon les besoins

- FAT : pas terrible, mais simple et rapide
- NTFS : Sécurisé, journalisé
- Ext, Ext2, Ext3, Ext4, ReiserFS, etc... Sous linux, sécurisé, inodes, etc...
- HFS+, HFSX pour Apple
- BtrFS (extend, snapshot)
- ISO 9660, UDF (cd, dvd, etc)

SGF POSIX: gestion interne

- Utilise souvent un ensemble de tables
- Un bitmap des blocs (ou d'un sur ensemble, tels les clusters) libres ou occupés
- Un bitmap des inodes (pointeurs sur les données, et méta données du fichiers)
- La table des inodes

Rôles des tables POSIX

- Bitmap des **clusters** : coche les *blocs* occupés : permet l'ajout de données, et le contrôles
- Bitmap des **inodes** : Coche les **inodes** déjà utilisées. Pour ajout et suppression
- Table des **inodes** : Quand on a **l'inode**, on trouve les droits, les dates accès, puis les *blocs* de données de **l'inode**

Inodes POSIX

- Contient toutes les métadonnées du fichier (SAUF LE NOM !!)
- Puis la liste des 11 premiers *blocs* de données
- Puis une indirection de *blocs* de données
- Puis une indirection d'indirections de *blocs* de données
- Puis une triple indirection !!
- Taille max d'un fichier : 17 Go

Le répertoire (ou dossier)

- Liste chaînées associant des noms à des **inodes**.
- Accéder à des données sous-entend :
 - Trouver le répertoire
 - Récupérer **l'inode** d'après le nom du fichier
 - Récupérer les données en suivant les infos de **l'inode**

Ajouter un fichier

- Trouver le répertoire.
- Avoir le droit de modifier ce répertoire.
- Créer une **inode** (si dispo)
- Créer les blocs et y mettre les données (si dispo)
- Bien mettre à jour les *tables*.
- Ajouter le nom du fichier et **l'inode** dans le répertoire

Supprimer un fichier

- Droit de modifier le répertoire (retirer un maillon de la liste chaînée)
- Si c'est le dernier pointeur sur **l'inode** :
 - Indiquer **l'inode** comme étant libre dans le bitmap des **inodes**
 - Indiquer tous les *blocs* comme étant libre dans le bitmap des *blocs*

SuperBlock

- Enfin, le **superbloc** contient une copie de toutes ces données et des informations liées au volume. Il y a des copies de ce **superbloc** a plusieurs endroits du volume.
- En cas de gros problème, il est possible de lire le disque en utilisant une des copies (si l'originale n'est plus lisible)

Le MONTAGE

- Toutes ces opérations sont complexes et mettent en péril l'organisation existante
- Il faut demander au prog spécialisé dans la gestion des **FS** de prendre en compte d'un support (ou d'arreter de le faire)
- On appelle cette opération le **montage** ou le **démontage**

Le montage

- Prise en compte de l'organisation du **FS** d'un support
- Accès à celui-ci via l'arborescence
- On **monte** donc un volume dans un répertoire (*/ si c'est le disque principal, ou sinon : /home, /var, etc...*)
- Pour des **montages** occasionnels, on a */media* ou */mnt*
- Les montages usuels sont décrits dans */etc/fstab*

Mount

- Plein d'options (lecture seule, à la volée (*automount*), accepte des exécutables ou pas, mise à jour synchrone ou asynchrone, accès aux users ou pas...)
- Permet le **montage** de volume (*cd*, *partition*, *clé usb*) ou de fichiers (**loop**)
- Permet des **montages** distants (*NFS*, *SMB*, *ssh*)

Montage suite

- *Mount* seul donne les montages actifs
- Le démontage se fait par *umount*
- */etc/fstab* décrit les montages connus, à faire automatiquement, dès le démarrage (ou pas)..
- Le nom des devices peut être soit dans */dev*, soit des noms stockés dans le FS
- Surveiller */var/log/syslog* et *lshw*

/etc/fstab

```
# <file system><mountpoint><type><options><dump><pass>  
proc      /proc      proc      nodev,noexec,nosuid 0      0
```

```
# / was on /dev/sda5 during installation
```

```
UUID=75...aa2d /          ext4      errors=remount-ro 0      1
```

```
# swap was on /dev/sda6 during installation
```

```
UUID=bab-... 282 none          swap      sw          0      0
```



Administration des systèmes

2013

19/19