

# + Module Linux

Equipe Pédagogique

S. BEN YAALA

H. SLIMANI



## Gestion des disques

# + Plan : Gestion des disques

- Partitionnement
- Système de fichiers
- Montage

# + Définition de partitionnement

- Le partitionnement consiste à créer une ou plusieurs zones de stockage indépendantes et de plus ou moins grandes tailles sur un disque dur. Ces zones sont appelées partitions.
- Chaque partition ainsi créée est gérée par le système comme un disque dur indépendant, même si physiquement il n'en existe qu'un seul.
- Chaque disque dur doit au moins posséder une partition .

# + Intérêt de partitionnement

- Installer plusieurs systèmes d'exploitation
- Séparer le système et les données
- Incompatibilité de certains systèmes

# + Définition d'une partition

- Une partition est un espace destiné à recevoir des fichiers via un système de fichier
- Un disque dur peut contenir plusieurs partitions.
- Le nombre de partitions est fonction du type de [table de partitionnement](#)

# + Partitions / et swap

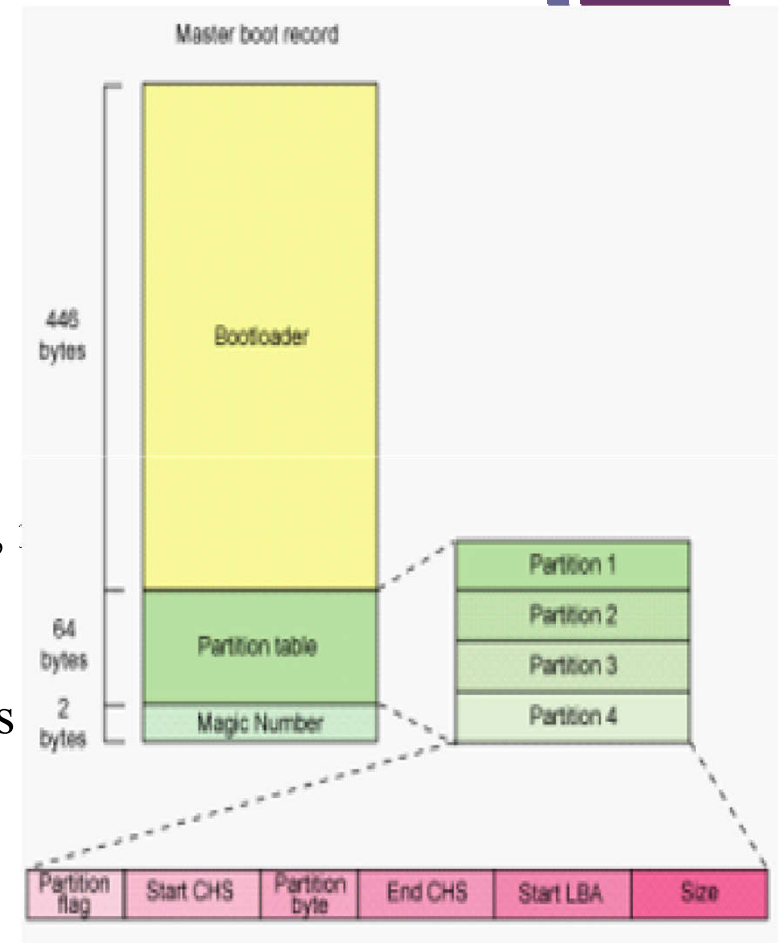
- Partition / : Elle contient la distribution Linux.
- Partition swap : elle permet d 'éxecuter plus de processus que la RAM seule ne pourrait en supporter.

# + Table de partitionnement

- La table de partitionnement se trouve dans le premier secteur accessible d'un disque dur physique. Elle contient les informations concernant les partitions d'un disque dur.
- Il existe deux tables de partitionnement : celle du **MBR** (Master Boot Record) et le **GPT** (GUID Partition Table).

# + Le MBR – Master Boot Record

- Le Master Boot Record ou MBR « zone amorce »
- Le MBR a une taille de 512 octets
- Le MBR contient principalement 2 éléments :
  - Le programme de boot que le BIOS lancera
  - La table des partitions
- La table des partitions principale contient 4 descripteurs, est donc limitée à 4 partitions
- On peut créer une partition étendue parmi les 4 partitions principales et créer de nouvelles partitions logiques à l'intérieur



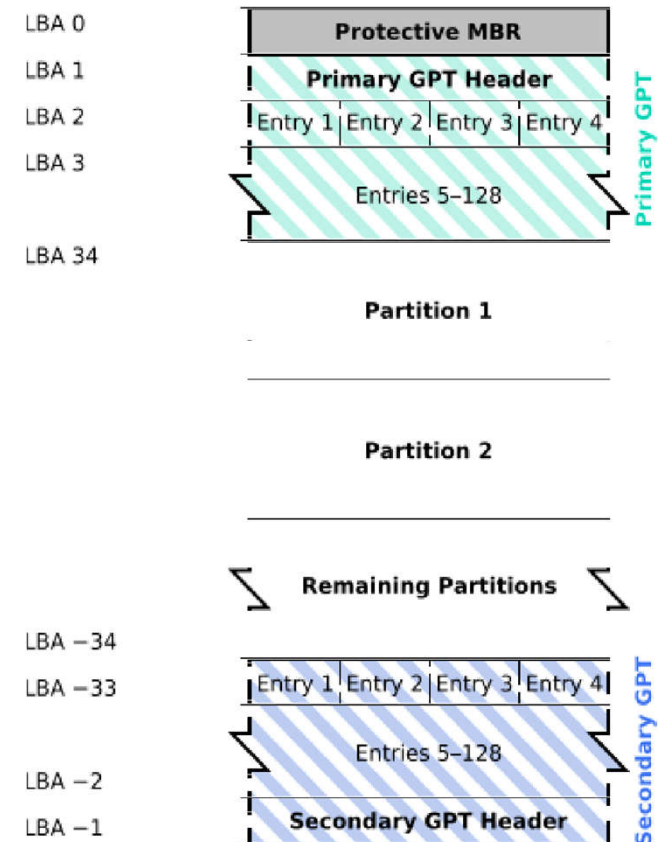


# + GPT -GUID Partition Table

- Il fait partie du standard EFI Extensible Firmware Interface
- Les informations concernant la table de partitionnement sont stockées dans un entête GPT, mais conserve une entrée MBR
- Il y a deux GPT sur le disque dur, l'un primaire, l'autre secondaire (sauvegarde du premier)
- GPT gère les disques durs et partitions jusqu'à 2 Zo (2,2 To en MBR)
- GPT permet presque un nombre illimité de partitions, la limite est la taille de la table de partition, par défaut 128 partitions

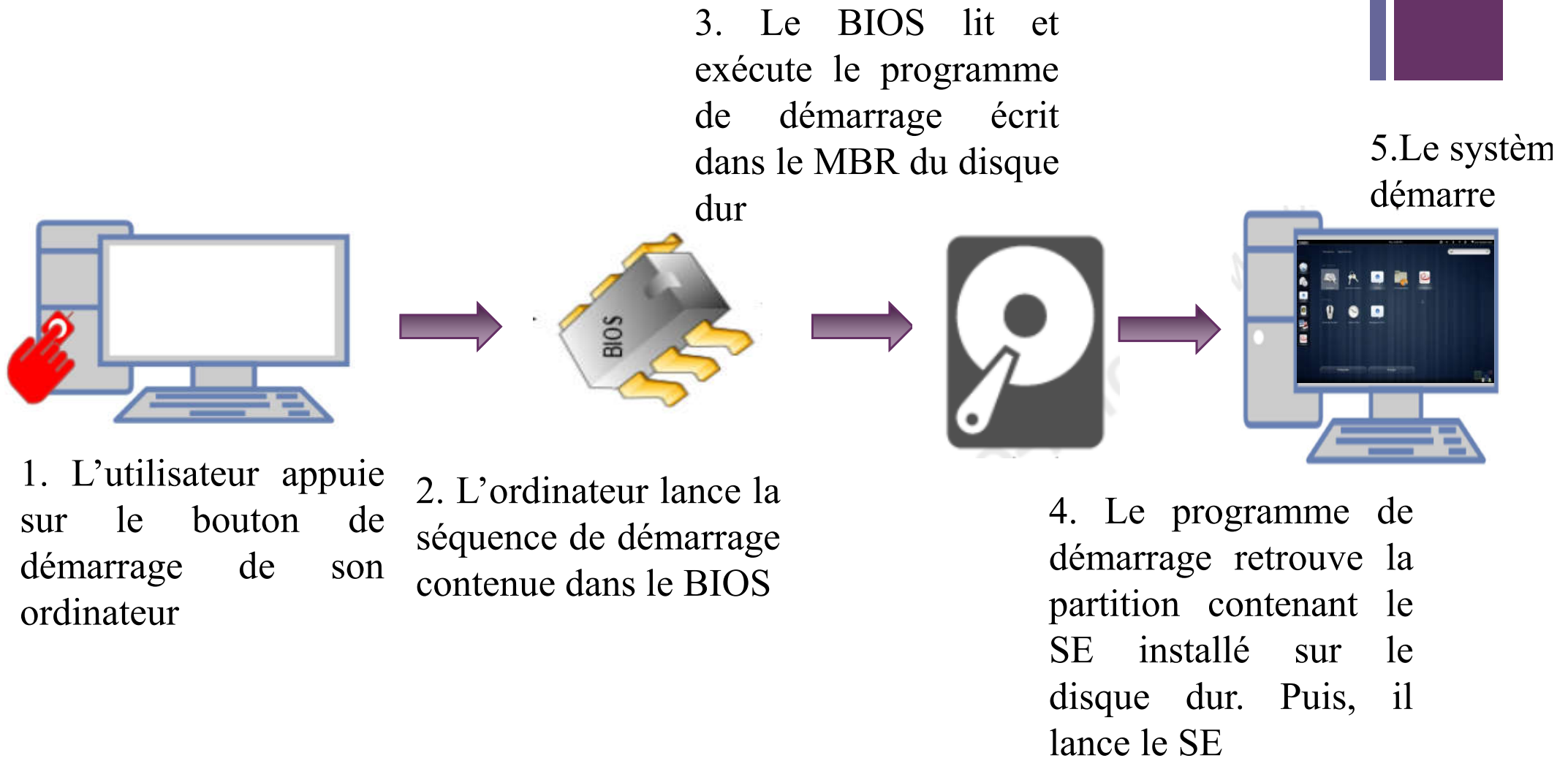


## GUID Partition Table Scheme



# + Le boot et le rôle du BIOS et du MBR

10



Concernant le rôle de l'UEFI et du GPT, il suffit de remplacer BIOS par UEFI, et MBR par GPT

# + Comparaison entre BIOS et UEFI

- BIOS peut lire la table de partitionnement MBR
- BIOS ne peut pas lire la table de partitionnement GPT.
- UEFI peut lire les 2 types de tables de partitionnement.

# + Organisation du disque avec MBR

12

- Organisation typique

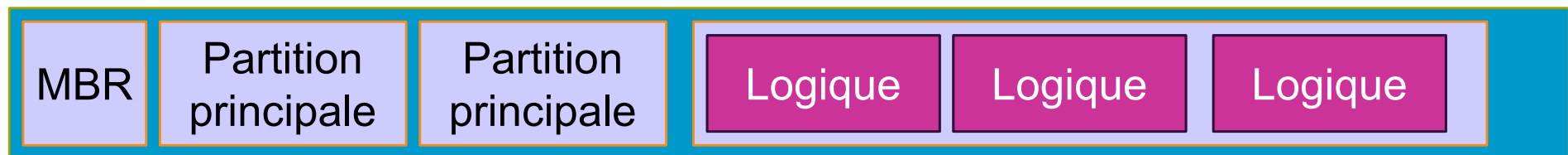


- Le **Master Boot Record** est situé dans les 1<sup>er</sup> secteurs du disque
- Il est constitué de 2 parties :
  - Table de partitions
  - Le programme d'amorçage qui charge le noyau du système
- Plusieurs types de partitions
  - Principale
  - Etendue
  - Logique

# + Les partitions avec MBR

13

- Les partitions principales
  - Au maximum de 4
  - Accepte tout type de système de fichiers
- Les partitions étendues
  - Destinées à contenir des partitions logiques et non un système de fichiers
  - Nécessitent au moins une partition principale
- Les partitions logiques
  - Contenues dans une partition étendue
  - Accepte tout type de système de fichiers
- Exemple permettant d'installer plusieurs systèmes d'exploitation



# + Prise en charge des disques sous Linux

14

- Le pointeur spécial /dev permet l'accès aux disques
  - Format des pointeurs sur disque :

Lettre de périphérique

**/dev/XYZ**

Type de bus N° de partition

- Types de bus
  - hd : Périphériques IDE
  - sd : Périphériques SCSI
  - sd : Périphériques SATA
- Exemples
  - /dev/hda1 :
    - Partition 1 sur le 1<sup>er</sup> disque IDE
  - /dev/sdb2 :
    - Partition 2 sur le 2<sup>ème</sup> disque Sata

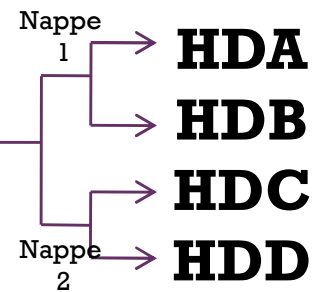
# +Nommage

15

**IDE**



**HD**

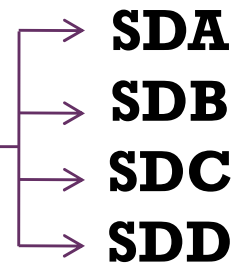


**HDA1**  
**HDA2**  
**HDA3**  
...

**SATA**



**SD**



**SDA1**  
**SDA2**  
...

# + Nommage des périphériques

- Si le disque dur est IDE, son nom commencera par hd.
- Si le disque est SATA ou SCSI ou USB, son nom commencera sd.
- Sur la première nappe :
  - Le disque maitre est **hda**
  - Le disque esclave est **hdb**
- Sur la deuxième nappe :
  - Le disque maitre est **hdc**
  - Le disque esclave est **hdd**
- Le premier disque sata est sda, le deuxième sdb, etc.
- Si le disque est partitionné, on ajoute seulement un numéro : sda1, hdb4.
- **La numérotation des partitions logiques commence à 5.**



# + Nommage des périphériques

17

## IDE SATA

### SATA

Linux <=2.4 /dev/hda  
linux >= 2.6 /dev/sda



## Disquette

### FLOPPY

/dev/fd0



## USB : Assimilé SCSI

### USB

/dev/sda

/dev/sdc

/dev/sdb



## IDE SCSI

### SCSI

Id : 0



/dev/sda

Id : 1



/dev/sdb

Id : 2



/dev/sdc

# + Les formats des systèmes de fichiers (1)

18

- À chaque système est associé un format
  - Définit la structure des données sur le support
- Sous Linux
  - ext2, ext3, ext4, jfs, xfs
  - ext3 et ext4 les plus courants pour Linux
- Sous Windows
  - fat, fat32, ntfs
  - Nfts est utilisé sous windows XP et Vista
- Toujours préférer un système de fichier « journalisé »
  - Chaque séquence de lecture/écriture est d'abord inscrite dans un journal avant d'être effectuée
    - Si le système se bloque pendant la séquence, elle sera achevée après le redémarrage
  - On évite les erreurs dans le système de fichiers

# + Les formats des systèmes de fichiers (2)

Systeme d'exploitation	Types de système de fichiers supportés
Dos	FAT16
Windows 95	FAT16
Windows 95 OSR2	FAT16, FAT32
Windows 98	FAT16, FAT32
Windows NT4	FAT, NTFS (version 4)
Windows 2000/XP	FAT, FAT16, FAT32, NTFS (versions 4 et 5)
Linux	Ext2, Ext3, ReiserFS, Linux Swap, (FAT, NTFS, ...)
MacOS	HFS, MFS
FreeBSD, OpenBSD	UFS (Unix File System)
BM AIX	JFS (Journaled File System)

# + Comparaison des formats des systèmes de fichiers (1)

Nom du système de fichiers	Taille maximale d'un fichier	Taille maximale d'une partition	Journalisée ou non ?	Gestion des droits d'accès?	Notes
<b>ext2fs</b> (Extended File System)	2 TiB	4 TiB	Non	Oui	Extended File System est le système de fichiers natif de Linux. En ses versions 1 et 2, on peut le considérer comme désuet, car il ne dispose pas de la journalisation. Ext2 peut tout de même s'avérer utile sur des disquettes 3½ et sur les autres périphériques dont l'espace de stockage est restreint, car aucun espace ne doit être réservé à un journal.
<b>ext3fs</b>	2 TiB	4 TiB	Oui	Oui	ext3 est essentiellement ext2 avec la gestion de la journalisation. Il est possible de <b>passer une partition formatée en ext2 vers le système de fichiers ext3 (et vice versa)</b> sans formatage.
<b>ext4fs</b>	16 TiB	1 EiB	Oui	Oui	ext4 est le successeur du système de fichiers ext3. Il est cependant considéré par ses propres concepteurs comme une solution intérimaire en attendant le vrai système de nouvelle génération que sera Btrfs.
<b>ReiserFS</b>	8 TiB	16 TiB	Oui	Oui	Développé par Hans Reiser et la société Namesys, ReiserFS est reconnu particulièrement pour bien gérer les fichiers de moins de 4 ko. Un avantage du ReiserFS, par rapport à ext3, est qu'il ne nécessite pas une hiérarchisation aussi poussée: il s'avère intéressant pour le stockage de plusieurs fichiers temporaires provenant d'Internet. Par contre, ReiserFS n'est pas recommandé pour les ordinateurs portables, car le disque dur tourne en permanence, ce qui consomme beaucoup d'énergie.

# + Comparaison des formats des systèmes de

Nom du système de fichiers	Taille maximale d'un fichier	Taille maximale d'une partition	Journalisée ou non ?	Gestion des droits d'accès?	Notes
<b>FAT</b> (File Allocation Table)	2 GiB	2 GiB	Non	Non	Développé par Microsoft, ce système de fichiers se rencontre moins fréquemment aujourd'hui. Il reste néanmoins utilisé sur les disquettes 3½ formatées sous Windows et devrait être utilisé sous Linux si une disquette doit aussi être lue sous Windows. Il est aussi utilisé par plusieurs constructeurs comme système de fichiers pour cartes mémoires ( <i>memory sticks</i> ), car, bien documenté, ce système de fichiers reste le plus universellement utilisé et accessible.
<b>FAT32</b>	4 GiB	8 TiB	Non	Non	Ce système de fichiers, aussi créé par Microsoft, est une évolution de son prédécesseur. Depuis sa version XP, Windows ne peut pas (ou bloque volontairement) formater une partition en FAT32 d'une taille supérieure à 32 Go. Cette limitation ne s'applique pas sous Linux, de même qu'avec des versions antérieures de Windows. Une partition FAT32 d'une taille supérieure à 32 Go formatée pourra être lue par Windows
<b>NTFS</b> (New Technology File System)	Limitée par la taille de la partition	2 TiB	Oui	Non	Ce système de fichiers a aussi été développé par Microsoft, et il reste très peu documenté. L'écriture depuis Linux sur ce système de fichiers a longtemps été considéré comme risqué; aujourd'hui, on dit que le support de l'écriture est stable. Néanmoins, puisque peu de tests ont encore été expérimentés, il est plus prudent de dire que <b>l'écriture sur ce type de partition doit être considéré comme risquée et expérimentale</b> . Pour autoriser l'écriture sur une partition NTFS sous Ubuntu, vous devrez activer un module du noyau ou utiliser le pilote Captive-ntfs

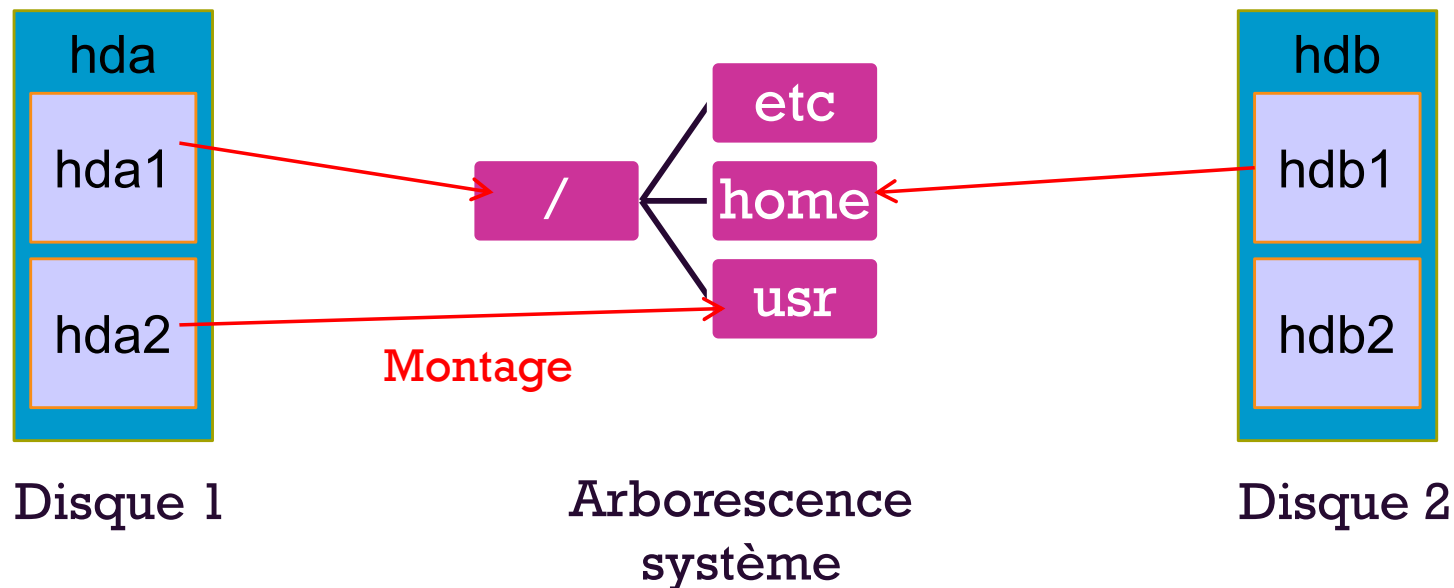
# + Les formats des systèmes de fichiers (3)

- Linux peut lire la plupart des formats
  - Notamment Ceux de Windows : NTFS, FAT, FAT32
- Avant d'être utilisé, un disque doit être partitionné
  - A l'aide des commandes (fdisk, gdisk,.. ) si Linux est déjà installé
  - Par le programme d'installation sinon (dépend de la distribution)
- Il faut ensuite créer un système de fichier
  - Avec l'utilitaire générique « mkfs »
    - mkfs.ext2, mkfs.ext3, mkfs.xfs, ...

# + Points de montage (1)

23

- Sous Linux, « Tout est fichier »
  - L'arborescence est construite à partir de « points de montage »
- Un point de montage est une association entre une partition physique et l'arborescence du système



- Avantages
  - Mettre à l'abri certaines données stratégiques comme /home
  - La défaillance du disque hdb n'entraîne pas une réinstallation totale

## + Points de montage (2)

- Tant qu'ils ne sont pas effectués, le système de fichiers est inaccessible
- Ils sont réalisés automatiquement au démarrage du système
  - Seulement ceux qui figurent dans le fichier « **/etc/fstab** »
- Il est possible de créer un point de montage manuellement
  - Pour les clés USB ou le CDROM par exemple
  - En utilisant la commande « mount »
  - Et « umount » pour supprimer le point de montage

```
mount /dev/hdd /mnt/cdrom  
umount /dev/hdd
```

- Une partition est associée à un système de fichiers
  - Il faut parfois préciser le type de ce système
    - ext2, ext3, xfs, swap, jfs, iso9660, vfat, ...



# + Monter un système de fichiers

- **Monter un système de fichier**

```
#mount -t xfs /dev/sdb1 /mnt
```

- **Visualiser les fs montés**

```
#mount
```

# + Le fichier /etc/fstab

- Le fichier /etc/fstab (File System Table) est le fichier qui contient les fs à monter lors du démarrage de la machine.
- Il contient donc la table des systèmes de fichiers à monter :

```
# /etc/fstab
#
/dev/mapper/rhel-root / xfs defaults 0 0
UUID=56fb2f30-d966-40d6-ab5a-0897aadc9ee2 /boot xfs defaults 0 0
/dev/mapper/rhel-swap swap swap defaults 0 0
```

# + Partitions MBR avec fdisk

- 3 étapes sont nécessaires
  - Partitionnement: Création d'une partition : **fdisk**
  - Formatage : Initialisation du système de fichiers : **mkfs**
  - Montage : Création d'un point de montage : **mount**

# + fdisk

- Manipuler la table de partitions d'un disque en mode interactif:

`fdisk <disque>`

- Afficher les partitions d'un disque :

`fdisk -l <disque>`

# + fdisk – type de partition

29

## ■ Code hexadécimal: Exemples :

- 5 : Partition étendue.
- 7 : NTFS.
- b : FAT32
- 82 : Linux swap
- 83 : Linux

# + fdisk - Commandes

- m : afficher l'aide.
- p : lister les partitions.
- n : créer une nouvelle partition.
- d : supprimer une partition.
- t : modifier le type d'une partition.
- w : sauvegarder la table de partitions.

# + Créer un système de fichiers (formatage)

31

■ `mkfs -t <type> <partition>`

■ Le type détermine la commande à exécuter :

– ext3 : `mkfs.ext3`

– ext4 : `mkfs.ext4`

– reiserfs : `mkfs.reiserfs`

– vfat : `mkfs.vfat`

– ntfs : `mkfs.ntfs`

● Exemples :

```
# mkfs -t ext3 /dev/sda1
```

```
# mkfs.ext3 /dev/sda1
```

```
# mke2fs -j /dev/sda1
```

● Transformer une partition ext2 en ext3 :

```
# tune2fs -j /dev/sda2
```



## *Exemple :*

Création d'une partition principale de 100Mo avec « fdisk » sur  
/dev/sdb



# + Etape 1 : Création de la partition

33

```
[root@localhost ~]# fdisk /dev/sdb
Welcome to fdisk (util-linux 2.23.2).

Changes will remain in memory only, until you decide to write them.
Be careful before using the write command.

Device does not contain a recognized partition table
Building a new DOS disklabel with disk identifier 0x96582d79.

Command (m for help): n
Partition type:
   p   primary (0 primary, 0 extended, 4 free)
   e   extended
Select (default p): p
Partition number (1-4, default 1): 1
First sector (2048-10485759, default 2048):
Using default value 2048
Last sector, +sectors or +size{K,M,G} (2048-10485759, default 10485759): +100M
Partition 1 of type Linux and of size 100 MiB is set

Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.
Syncing disks.
```

## + Etape 2: Initialisation du système de fichiers

```
[root@localhost ~]# mkfs.ext3 /dev/sdb1
mke2fs 1.42.9 (28-Dec-2013)
Filesystem label=
OS type: Linux
Block size=1024 (log=0)
Fragment size=1024 (log=0)
Stride=0 blocks, Stripe width=0 blocks
25688 inodes, 102400 blocks
5120 blocks (5.00%) reserved for the super user
First data block=1
Maximum filesystem blocks=67371008
13 block groups
8192 blocks per group, 8192 fragments per group
1976 inodes per group
Superblock backups stored on blocks:
    8193, 24577, 40961, 57345, 73729

Allocating group tables: done
Writing inode tables: done
Creating journal (4096 blocks): done
Writing superblocks and filesystem accounting information: done
```

# +Etape 3 :Création du point de montage

35

- Création du point de montage
  - De type ext3
  - Avec la commande « mount »

```
[root@localhost ~]# mount -t ext3 /dev/sdb1 /mnt/toto
```

- Vérification du point de montage avec la commande « df »

```
[root@localhost ~]# df
```

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
/dev/sda3	18555904	4502624	14053280	25%	/
devtmpfs	485252	0	485252	0%	/dev
tmpfs	499968	100	499868	1%	/dev/shm
tmpfs	499968	7204	492764	2%	/run
tmpfs	499968	0	499968	0%	/sys/fs/cgroup
/dev/sda1	303780	154912	148868	51%	/boot
tmpfs	99996	12	99984	1%	/run/user/1000
/dev/sdb1	95054	1567	88367	2%	/mnt/toto

# + partition d'échange (swap) Linux

36

- Créer une partition d'échange (swap) Linux.

`mkswap <partition>`

- Activer le swap :

`swapon <partition>`

- Désactiver le swap :

`swapoff <partition>`

- Le fichier **/proc/swaps** contient la liste des zones de swap actives.

# + Autres outils

- gdisk : Équivalent à fdisk mais spécifique au partitionnement GPT.
- parted : Outil avec un haut niveau d'abstraction (plus simple).
- gparted : Interface graphique pour parted.

# + Partitions GPT avec gdisk

```
[root@server1 ~]# gdisk /dev/vdc
GPT fdisk (gdisk) version 0.8.6
```

```
Partition table scan:
```

```
MBR: not present
BSD: not present
APM: not present
GPT: not present
```

```
Creating new GPT entries
```

```
Command (? for help): n
```

```
Partition number (1-128, default 1): 1
```

```
First sector (34-2097118, default = 2048) or {+-}size{KMGTP}: 2048
```

```
Last sector (2048-2097118, default = 2097118) or {+-}size{KMGTP}:  
+500M
```

```
Current type is 'Linux filesystem'
```

```
Hex code or GUID (L to show codes, Enter = 8300):
```

```
Changed type of partition to 'Linux filesystem'
```

```
Command (? for help): w
```

# + Partitionnement avec parted

- **Lancer l'outil parted**

```
root@localhost ~# parted
```

- Sélection du disque et création de la table de type GPT

```
(parted) select /dev/sdb
```

```
(parted) mklabel New disk label type? [msdos]? gpt
```

- Création de la partition

```
(parted) mkpart
```

```
Nom de la partition ? []? Var
```

```
Type de système de fichiers ? [ext2]? xfs
```

```
Début ? 50G
```

```
Fin ? 60G
```

```
Numéro Début Fin Taille Système de fichiers Nom
```

```
3 50,0GB 60,0GB 9999MB var
```

# + Partitionnement avec Gparted

40

