



---

# Administration & Sécurité des Systèmes d'Exploitation

---

Année Universitaire  
2022-2023

---

# Chapitre IV : Gestion des disques

---

---

# Objectifs

---

- Concevoir le plan de partitionnement des disques d'un système Linux.
- Formater des partitions et Gérer les systèmes de fichiers.
- Contrôler le montage et le démontage d'un système de fichiers.
- Gérer les quotas disque.

---

# Plan

---

- Introduction : Concepts de Base
- FHS : File Hierarchy Standard
- Système de Gestion de Fichiers
- Partitionnement des disques
- Gestion des quotas

---

# Introduction : Concepts de Base

---

---

# Fichier

---

- **Fichier sous Linux :**
  - est désigné par un nom
  - possède une unique **inode** (contient certaines informations concernant le fichier).
  - Possède les fonctionnalités suivantes :
    - Ouverture
    - Fermeture
    - Lecture (consultation)
    - Ecriture (modification)
  - Peut être de type :
    - ordinaire (ou "normal") (-)
    - répertoire (**d**)
    - lien symbolique (**l**)

---

# Inode

---



- Est une **structure de données** contenant des informations concernant les fichiers stockés dans les systèmes de fichiers
- À chaque fichier correspond une **inode unique** avec un **numéro d'inode unique** dans le système de fichiers dans lequel il réside.

---

# Inode

---

- Les inodes peuvent, selon le système de fichiers, contenir des informations concernant le fichier:
  - Numéro d'inode,
  - Propriétaire,
  - Groupe propriétaire,
  - Droits d'accès,
  - Taille en octets,
  - Type,
  - Horodatage: date de création, date de modification
  - Nombre de liens physiques, etc



---

# Inode

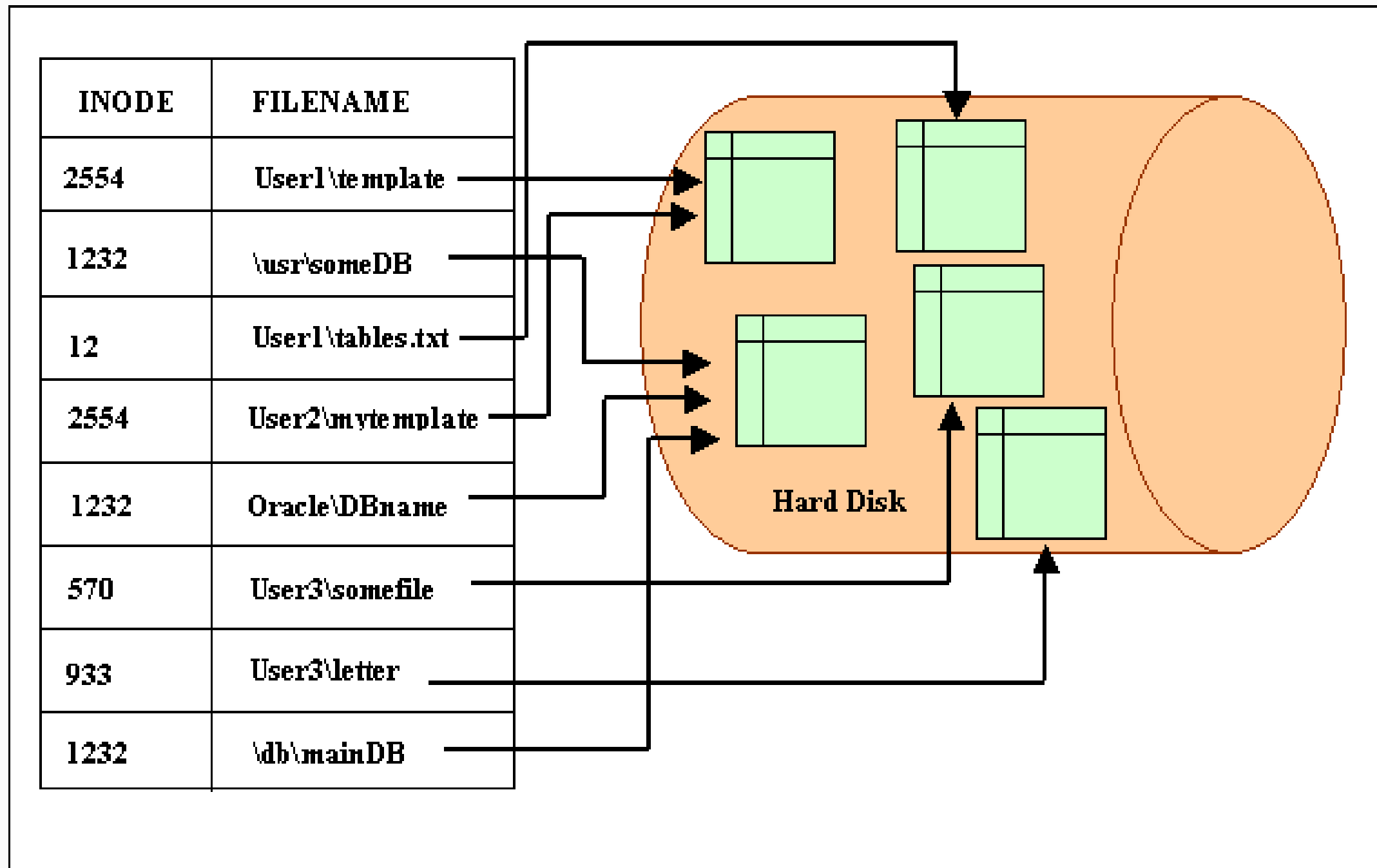
---

- Les inodes contiennent notamment les métadonnées des fichiers, et en particulier celles concernant les droits d'accès.
- Pour afficher le numéro d'inode d'un fichier « toto », on utilise la commande.

```
esprit@esprit-virtual-machine:~$ ls -li toto  
1080971 toto
```

- La quantité d'inodes (généralement déterminée lors du formatage et dépendant de la taille de la partition) indique le nombre maximum de fichiers que le système de fichiers peut contenir.

# Inode



---

# Inode

---

- La commande **ls -i** affiche le numéro d'inode d'un fichier

```
esprit@esprit-virtual-machine:~$ ls -i fichier
1081048 fichier
```

- La commande **stat** affiche des informations détaillées sur un fichier.

```
esprit@esprit-virtual-machine:~$ stat fichier
  File: fichier
  Size: 0          Blocks: 0          IO Block: 4096   regular empty file
Device: 803h/2051d Inode: 1081048    Links: 1
Access: (0664/-rw-rw-r--)  Uid: ( 1000/   esprit)   Gid: ( 1000/   esprit)
Access: 2022-10-24 11:20:19.331996303 +0100
Modify: 2022-10-24 11:20:19.331996303 +0100
Change: 2022-10-24 11:20:19.331996303 +0100
 Birth: 2022-10-24 11:20:19.331996303 +0100
```

---

# Inode

---

## Commandes *ls -i* & *df -i*

```
esprit@esprit-virtual-machine:~$ ls -i fichier
1081048 fichier
```

```
esprit@esprit-virtual-machine:~$ df -i /dev/sda3
Filesystem      Inodes   IUsed   IFree IUse% Mounted on
/dev/sda3      1277952 259869 1018083   21% /
```

## Commandes *stat*

```
esprit@esprit-virtual-machine:~$ stat --file-system fichier
File: "fichier"
  ID: 5965af4e08675d58 Namelen: 255      Type: ext2/ext3
Block size: 4096      Fundamental block size: 4096
Blocks: Total: 4986524   Free: 1599165   Available: 1339530
Inodes: Total: 1277952   Free: 1018083
```

---

FHS

---

---

# FHS

---

- **FHS : File Hierarchy System** définit l'arborescence (la répartition logique des fichiers), et le contenu des principaux répertoires des systèmes de fichiers des systèmes d'exploitation GNU/Linux.
- FHS a pour objectif de proposer la **normalisation** de l'organisation de système de fichiers pour les systèmes Unix.
- FHS définit les noms, emplacements, et permissions pour de nombreux types de fichiers et répertoires.

---

# FHS

---

Répertoire	Description
bin	Programmes utilisateur essentiels (nécessaires au démarrage du système)
boot	Fichiers nécessaires au chargement de Linux (bootloader, initrd, noyau)
dev	Fichiers spéciaux offrant l'accès aux périphériques
etc	Configuration du système et des services
home	Répertoires personnels de tous les utilisateurs
lib	Librairies partagées essentielles (nécessaires au démarrage du système)
mnt	Points de montage temporaires (cdrom, floppy, etc.)
proc	Systèmes de fichiers virtuels permettant d'accéder aux structures internes du noyau
root	Répertoire principal de l'utilisateur root
sbin	Exécutables système essentiels (nécessaires au démarrage du système)
tmp	Stockage de fichiers temporaires
usr	Arborescence contenant la plupart des fichiers des applications
var	Données vivantes du système et des applications
media	Point de montage d'un support amovible
opt	Paquetages des logiciels et des applications ajoutées



---

# Systeme de fichiers

---



---

# Système de fichiers

---

- Un système de fichiers est un programme qui permet de **stocker, lire et manipuler** des données sur un lecteur block (Disque dur, DVD, etc).
- Les systèmes de fichiers sont la seule manière efficace d'accéder au contenu des lecteurs block.
- Un système de fichiers maintient la structure interne des données (meta-data) qui fait que les données restent **organisées et accessibles**.

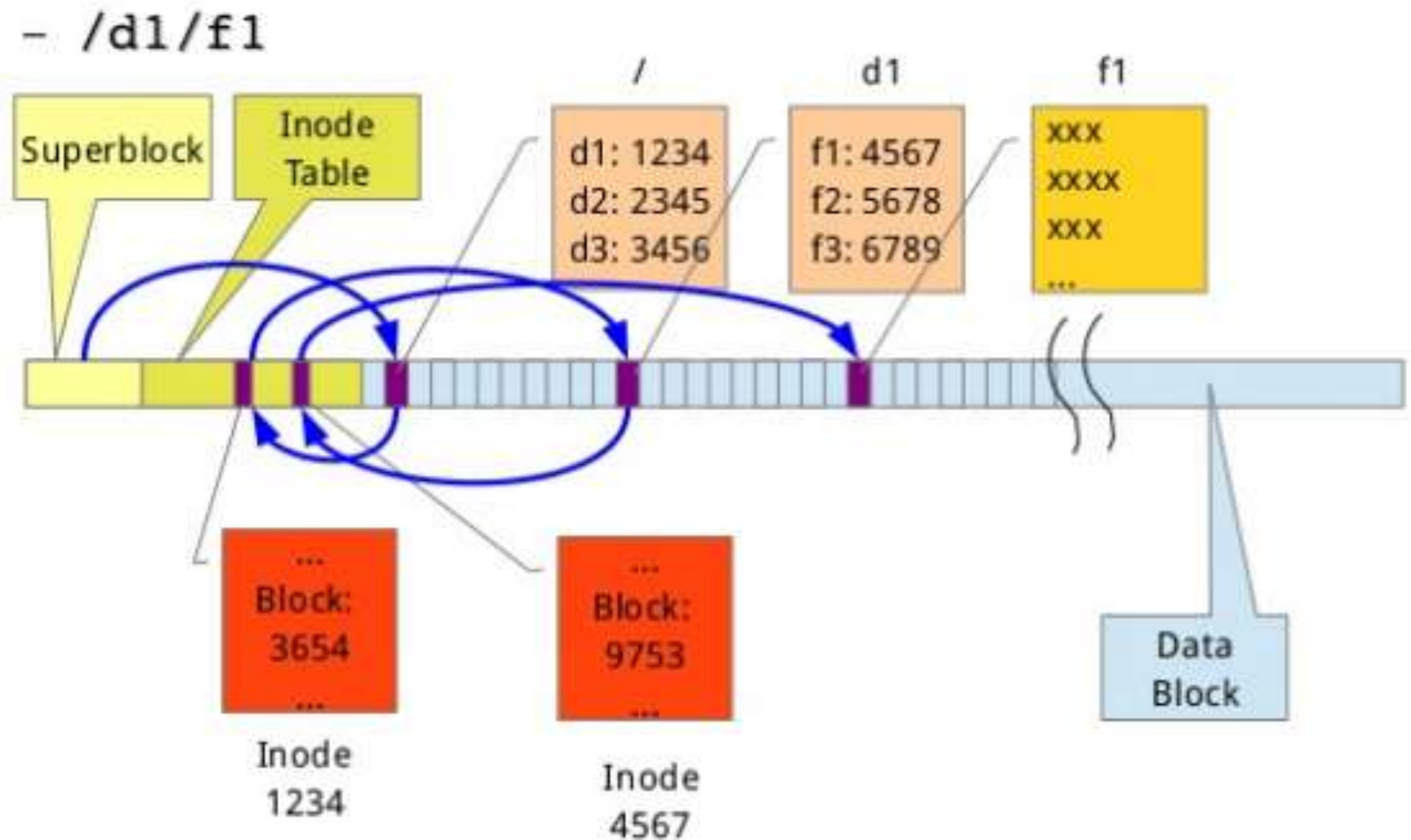
---

# Structure d'un système de fichiers

---

- La structure d'un système de fichiers est définie par trois entités :
  - **Superbloc** : contient toutes les informations concernant le système de fichier (meta-données) telles que la taille totale de système de fichier, le nombre de blocs libres, la taille d'un bloc de données, etc.
  - **Table des inodes** : contient la liste des inodes des fichiers.
  - **Blocs de données** : contiennent les fichiers réguliers (données) et répertoires.

# Structure d'un système de fichiers



---

# Types des systèmes de fichiers

---

- **ext2**: Ancien et très stable, il fonctionne pour des fichiers dont la taille est supérieurs à ~2-3K
- **ext3**: Extension journalisée de ext2
- **ext4**: Successeur de ext3
- **reiserfs**: Système de fichiers journalisé, plus performant pour les fichiers de petite taille (internet)
- **XFS**: Système de fichiers journalisé; permet d'optimiser la manipulation des très gros fichiers, de plus de 9 ExaBytes (9'000'000'000 GigaBytes)
- **JFS**: Système de fichiers journalisé créé par IBM
- **FAT32** et **NTFS**: Système de fichiers Windows

# Types des systèmes de fichiers

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</b> (et vice versa) 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.

---

# Journalisation

---

- Un **journal** est la partie d'un système de fichiers **journalisé** qui trace les opérations d'écriture tant qu'elles **ne sont pas terminées** et cela en vue de garantir l'intégrité des données en cas **d'arrêt brutal**.
- L'intérêt est de pouvoir plus facilement et plus rapidement récupérer les données en cas d'arrêt brutal du système d'exploitation (**coupure d'alimentation, plantage du système**, etc.), alors que les partitions n'ont pas été correctement synchronisées et démontées.

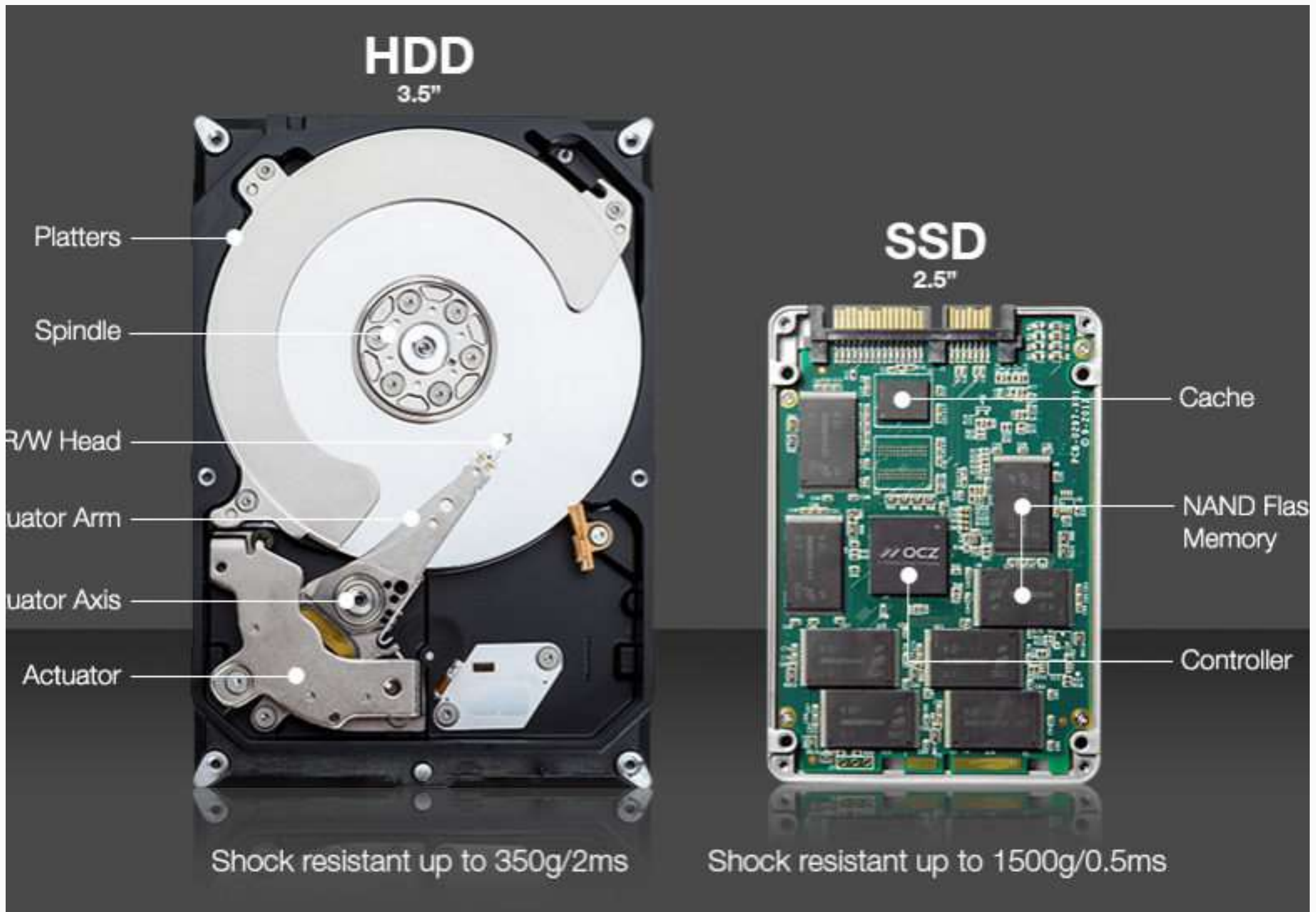
---

# Créer des partitions /Systèmes de fichiers

---



# Structure disque dur HDD Vs SSD





---

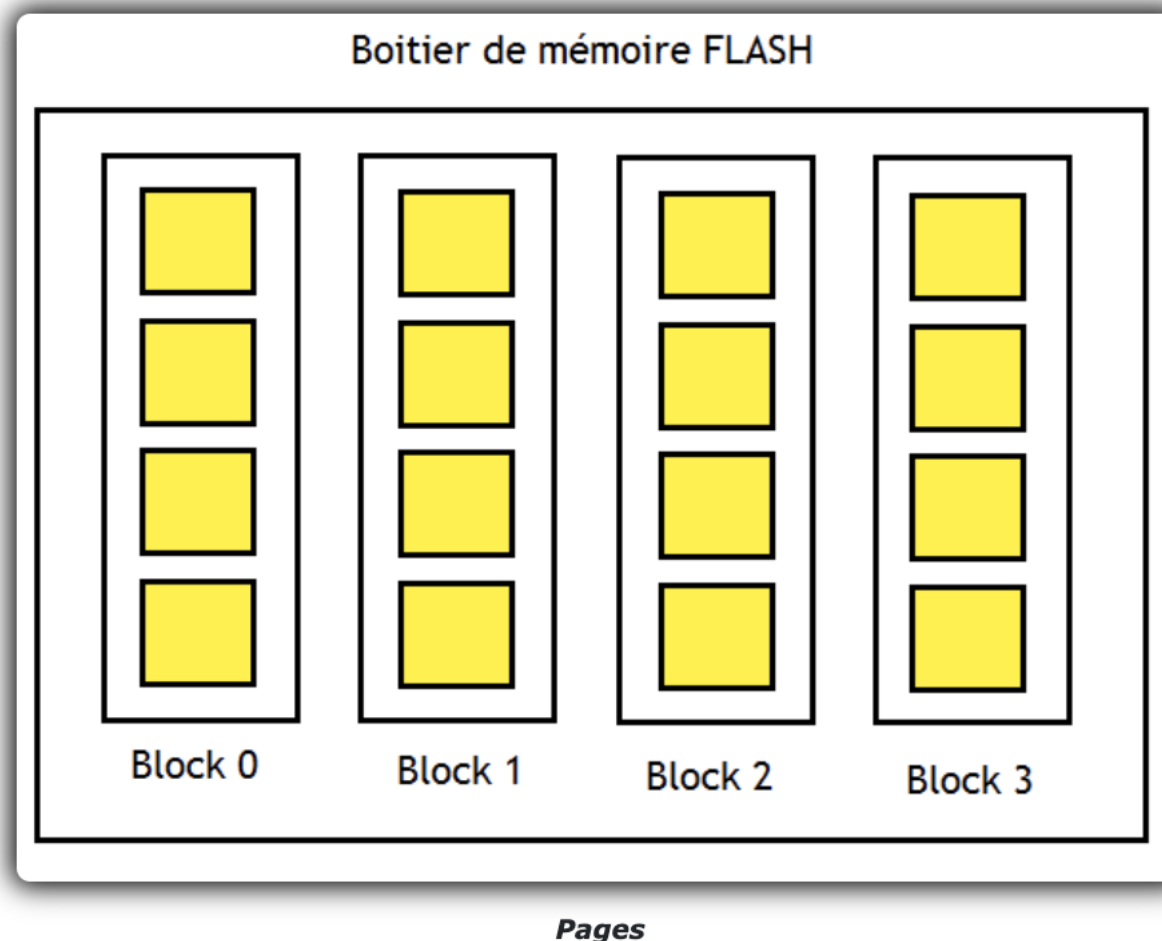
# Structure disque dur HDD Vs SSD

---

- Les disques durs HDD utilisent des plateaux magnétiquement sensibles qui sont déplacés par un moteur, ce qui les rend plus lent et plus fragiles.
- Les SSD utilisent une mémoire flash sans aucune pièce mobile, ce qui signifie qu'ils sont plus rapides.

# Structure disque dur HDD Vs SSD

- Les disques SSD se basent sur Les mémoires FLASH qui utilisent les pages et les blocs pour stocker et gérer les données.



---

# Définition Partition

---

- Une partition est une division logique d'un disque dur qui est traitée comme une unité distincte par les systèmes d'exploitation (OS) et les systèmes de fichiers.
- Le partitionnement est l'opération qui consiste à diviser ce support en partitions dans lesquelles le système d'exploitation peut gérer les informations de manière séparée en y créant un système de fichiers.

---

# Nommage des partitions sous Windows

---

- Nommage des partitions par des lettres de l'alphabet
- Commence à **C:** ---> **Z:**
- Pas d'information sur le numéro du disque dur
- Pas d'information sur le type de disque dur (IDE ou SATA)
- Pas s'information sur le type de partition

---

# Nommage des partitions sous Linux

---

- Si le disque dur est IDE, son nom commence par **hd**
- Si le disque dur est SATA ou SCSI ou USB, son nom commence par **sd**
- Sur la première nappe :
  - Le disque maître est **hda**
  - Le disque esclave est **hdb**
- Sur la deuxième nappe :
  - Le disque maître 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** pour indiquer le numéro de la partition

---

# Gestion des partitions

---

- Les outils fdisk, cfdisk, sfdisk, gdisk ou encore parted permettent de manipuler les partitions, sans compter les outils graphiques disponibles durant l'installation ou dans les panneaux de configuration.
- **fdisk** est le plus ancien et le plus utilisé des outils de partitionnement. Il n'a aucun rapport avec le fdisk de Microsoft. Il est à base de menus et raccourcis textuels.
- **gdisk** (GPT fdisk) est un outil de manipulation interactif de tables de partition GPT
- **cfdisk** est un peu plus « visuel » et s'utilise avec les flèches directionnelles. Il permet les mêmes opérations que fdisk mais de manière plus conviviale.
- **sfdisk** fonctionne en interactif ou non, est assez compliqué mais plus précis.
- **parted** permet des opérations très avancées sur les partitions comme par exemple leur redimensionnement.

# Gestion des partitions: *gdisk*

*Afficher la table de partitions GPT du périphérique indiqué*

```
esprit@esprit-virtual-machine:~$ sudo gdisk -l /dev/sda
[sudo] password for esprit:
GPT fdisk (gdisk) version 1.0.8

Partition table scan:
  MBR: protective
  BSD: not present
  APM: not present
  GPT: present

Found valid GPT with protective MBR; using GPT.
Disk /dev/sda: 41943040 sectors, 20.0 GiB
Model: VMware Virtual S
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): A750FD45-70AB-4142-B09D-BEEC5A7086AC
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 41943006
Partitions will be aligned on 2048-sector boundaries
Total free space is 4029 sectors (2.0 MiB)

Number  Start (sector)    End (sector)  Size      Code  Name
   1            2048             4095     1024.0 KiB   EF02
   2            4096          1054719     513.0 MiB   EF00  EFI System Partition
   3          1054720          41940991    19.5 GiB   8300
```

---

# Gestion des partitions: *gdisk*

---

- Lorsque gdisk démarre, il effectue une analyse de quatre types de tables de partition existantes et affiche les résultats :
- **MBR** est Master Boot Record partitioning system;
- **BSD** Berkeley Standard Distribution (BSD) utilisé sur certains ordinateurs qui exécutent un BSD (FreeBSD, OpenBSD, etc.);
- **APM** est Apple Partition Map utilisée sur les Macintosh 680x0 et PowerPC;
- **GPT** est la table de partition GUID.



---

# Gestion des partitions: *gdisk*

---

- Les analyses BSD et APM signalent qu'elles sont présentes ou non.
- GPT peut signaler trois états (présent, non présent ou endommagé)
- MBR peut signaler 4 états (MBR uniquement, protecteur, hybride ou non présent).

---

# Gestion des partitions : *gdisk*

---

*Créer des partitions en mode interactif*

```
esprit@esprit-virtual-machine:~$ sudo gdisk /dev/sdb
GPT fdisk (gdisk) version 1.0.8

Partition table scan:
  MBR: not present
  BSD: not present
  APM: not present
  GPT: not present

Creating new GPT entries in memory.

Command (? for help): █
```

---

# Gestion des partitions : *gdisk*

---

*Lister les options de gdisk*

```
Command (? for help): ?  
b      back up GPT data to a file  
c      change a partition's name  
d      delete a partition  
i      show detailed information on a partition  
l      list known partition types  
n      add a new partition  
o      create a new empty GUID partition table (GPT)  
p      print the partition table  
q      quit without saving changes  
r      recovery and transformation options (experts only)  
s      sort partitions  
t      change a partition's type code  
v      verify disk  
w      write table to disk and exit  
x      extra functionality (experts only)  
?      print this menu
```



# Gestion des partitions : *gdisk*

```
Command (? for help): l
Type search string, or <Enter> to show all codes:
0700 Microsoft basic data          0701 Microsoft Storage Replica
0702 ArcaOS Type 1                0c01 Microsoft reserved
2700 Windows RE                   3000 ONIE boot
3001 ONIE config                  3900 Plan 9
4100 PowerPC PReP boot            4200 Windows LDM data
4201 Windows LDM metadata         4202 Windows Storage Spaces
7501 IBM GPFS                     7f00 ChromeOS kernel
7f01 ChromeOS root               7f02 ChromeOS reserved
8200 Linux swap                  8300 Linux filesystem
8301 Linux reserved              8302 Linux /home
8303 Linux x86 root (/)           8304 Linux x86-64 root (/)
8305 Linux ARM64 root (/)         8306 Linux /srv
8307 Linux ARM32 root (/)         8308 Linux dm-crypt
8309 Linux LUKS                   830a Linux IA-64 root (/)
830b Linux x86 root verity        830c Linux x86-64 root verity
830d Linux ARM32 root verity      830e Linux ARM64 root verity
830f Linux IA-64 root verity      8310 Linux /var
8311 Linux /var/tmp               8312 Linux user's home
8313 Linux x86 /usr               8314 Linux x86-64 /usr
8315 Linux ARM32 /usr             8316 Linux ARM64 /usr
8317 Linux IA-64 /usr             8318 Linux x86 /usr verity
```

*Afficher les types et les codes de partitions qu'on peut créer*

```
ef00 EFI system partition          ef01 MBR partition scheme
ef02 BIOS boot partition           f800 Ceph OSD
f801 Ceph dm-crypt OSD             f802 Ceph journal
f803 Ceph dm-crypt journal         f804 Ceph disk in creation
f805 Ceph dm-crypt disk in creation f806 Ceph block
f807 Ceph block DB                 f808 Ceph block write-ahead log
f809 Ceph lockbox for dm-crypt keys f80a Ceph multipath OSD
f80b Ceph multipath journal        f80c Ceph multipath block 1
f80d Ceph multipath block 2        f80e Ceph multipath block DB
f80f Ceph multipath block write-ahead l f810 Ceph dm-crypt block
f811 Ceph dm-crypt block DB        f812 Ceph dm-crypt block write-ahead lo
f813 Ceph dm-crypt LUKS journal    f814 Ceph dm-crypt LUKS block
f815 Ceph dm-crypt LUKS block DB   f816 Ceph dm-crypt LUKS block write-ahe
f817 Ceph dm-crypt LUKS OSD        fb00 VMWare VMFS
```

# Types de partitions

Type de partition	Identifiant Globalement Unique (GUID) <sup>21</sup>	Code Hexadécimal
Non utilisé	00000000-0000-0000-0000-000000000000	
partition système <a href="#">EFI</a>	C12A7328-F81F-11D2-BA4B-00A0C93EC93B	ef00
schéma de partitionnement MBR	024DEE41-33E7-11D3-9D69-0008C781F39F	ef01
BIOS Boot partition	21686148-6449-6E6F-744E-656564454649	ef02
Linux filesystem	0FC63DAF-8483-4772-8E79-3D69D8477DE4	8300
Reserved	8DA63339-0007-60C0-C436-083AC8230908	8301
Linux /home	933AC7E1-2EB4-4F13-B844-0E14E2AEF915	8302
Linux /var	3B8F8425-20E0-4F3B-907F-1A25A76F98E8	8306
partition d'échange ( <i>swap</i> )	0657FD6D-A4AB-43C4-84E5-0933C84B4F4F	8200
partition <a href="#">gérée par volumes logiques</a> (LVM)	E6D6D379-F507-44C2-A23C-238F2A3DF928	8e00
tranche de volume RAID	A19D880F-05FC-4D3B-A006-743F0F84911E	fd00



---

# Partitions : *LVM*

---

- LVM (Logical Volume Manager) est à la fois une méthode et un logiciel de gestion de l'utilisation des espaces de stockage.
- La capacité de stockage traditionnelle est basée sur la capacité de chaque disque.
- LVM utilise un concept différent. L'espace de stockage est géré en combinant ou en mettant en commun la capacité des tous les disques disponibles.

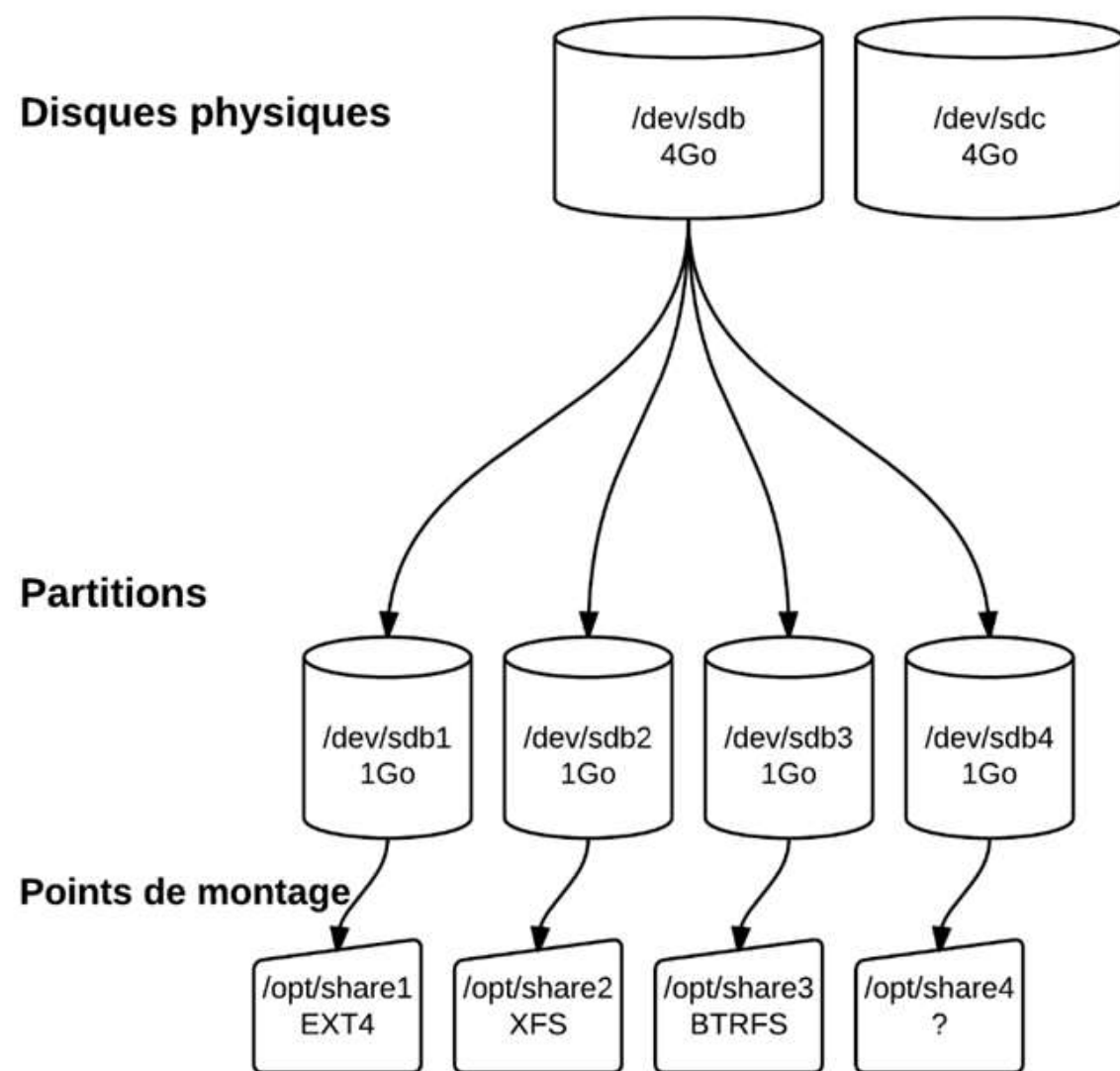
---

# Partitions : *LVM*

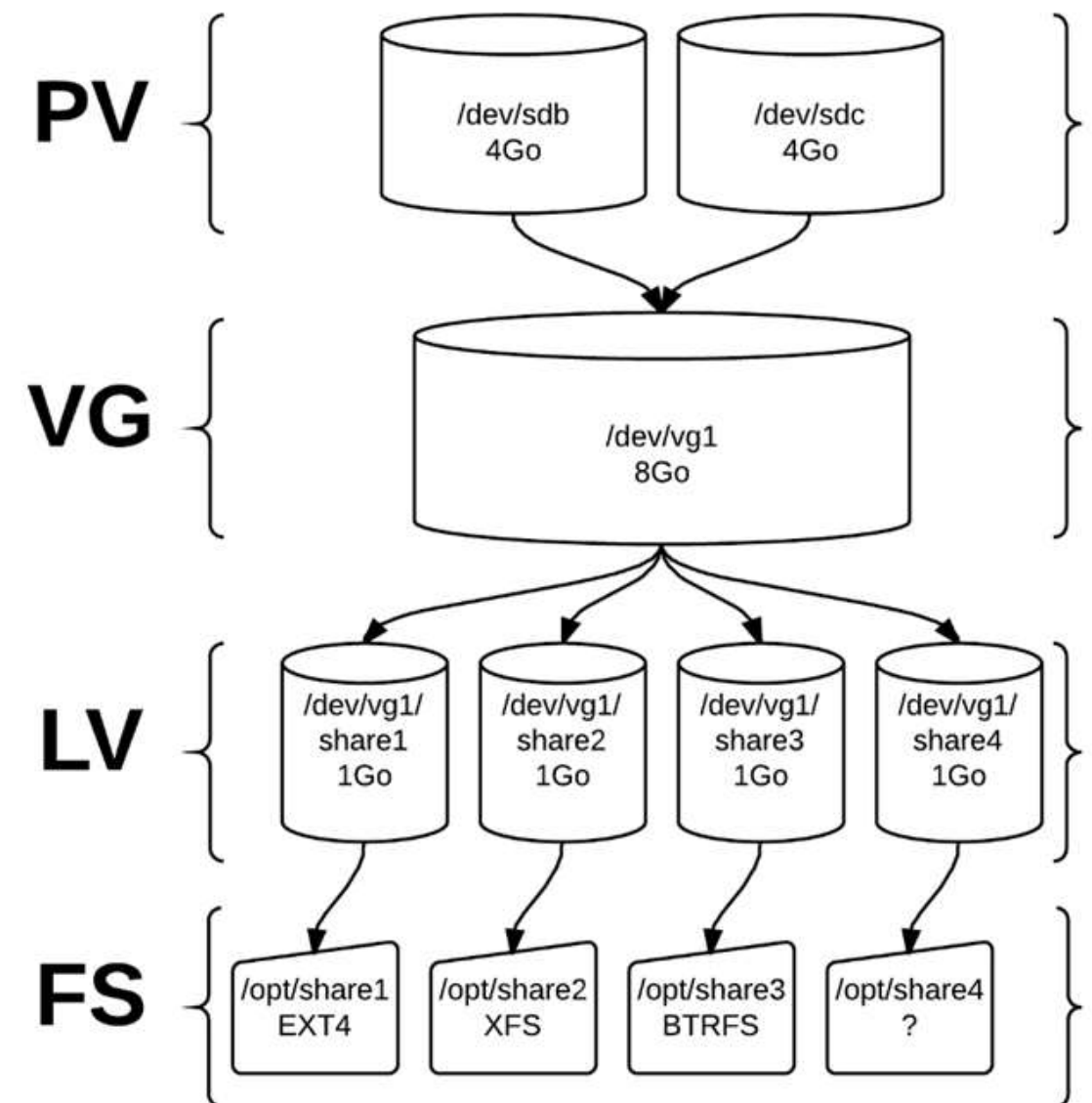
---

- Avec le stockage traditionnel, trois disques de 1 To sont gérés individuellement.
- Avec LVM, ces trois mêmes disques sont considérés comme une capacité de stockage agrégée de 3 To. Ceci est accompli en désignant les disques de stockage en tant que volumes physiques (PV) ou capacité de stockage utilisable par LVM.
- Les PV sont ensuite ajoutés à un ou plusieurs groupes de volumes (VG). Les VG sont découpés en un ou plusieurs volumes logiques (LV), qui sont ensuite traités comme des partitions traditionnelles.
- Le noyau voit les VG comme des périphériques de type block et leurs LV comme leurs partitions.

# Partitions : *LVM*



*Situation sans LVM*



*Situation avec LVM*



---

# Partitions : *LVM*

---

## ➤ **Avantages de LVM**

- Il n'y a pas de limitations « étranges » comme avec les partitions (primaire, étendue, etc.).
- On ne se préoccupe plus de l'emplacement exact des données.
- Les opérations de redimensionnement deviennent quasiment sans risques, contrairement au redimensionnement des partitions.
- On peut créer des snapshots de volume sans perturber le fonctionnement de la machine et sans interruption de services.

---

# Partitions : *LVM*

---

## ➤ Inconvénients de LVM

- Si un des volumes physique est endommagé , c'est l'ensemble des volumes logiques qui utilisent ce volume physique qui sont perdus.
- Pour éviter ce cas, il faudra utiliser LVM sur des disques **RAID**.

---

# Partition : *LVM-RAID*

---

- **RAID (Redundant Array of Independent Disks) :** RAID est un ensemble de techniques de virtualisation du stockage permettant de répartir des données sur plusieurs disques durs afin d'améliorer soit les performances, soit la sécurité ou la tolérance aux pannes de l'ensemble du ou des systèmes.

# Partitions : *LVM*

*Créer un volume physique à partir du disque dur*

```
esprit@esprit-virtual-machine:~$ sudo pvcreate /dev/sdc  
Physical volume "/dev/sdc" successfully created.
```

*Créer un volume groupe*

```
esprit@esprit-virtual-machine:~$ sudo vgcreate vg1 /dev/sdc  
Volume group "vg1" successfully created
```

*Afficher les propriétés du volume groupe*

```
esprit@esprit-virtual-machine:~$ sudo vgs  
--- Volume group ---  
VG Name                vg1  
System ID  
Format                 lvm2  
Metadata Areas         1  
Metadata Sequence No   1  
VG Access              read/write  
VG Status               resizable  
MAX LV                 0  
Cur LV                0  
Open LV                0  
Max PV                 0  
Cur PV                1  
Act PV                 1  
VG Size                <5.00 GiB  
PE Size                4.00 MiB  
Total PE               1279  
Alloc PE / Size        0 / 0  
Free PE / Size         1279 / <5.00 GiB  
VG UUID                JEUMh3-qDwc-Bquh-3Ddm-Uhwa-11QY-PhAE26
```

# Partitions : *LVM*

*Créer un volume logique nommé vol1 et de taille 1 GO*

```
esprit@esprit-virtual-machine:~$ sudo lvcreate -n vol1 -L 1g vg1  
Logical volume "vol1" created.
```

*Afficher les propriétés du volume logique*

```
esprit@esprit-virtual-machine:~$ sudo lvdisplay vg1  
--- Logical volume ---  
LV Path                /dev/vg1/vol1  
LV Name                 vol1  
VG Name                 vg1  
LV UUID                 W11Le6-5TJ7-GSF6-0eu1-of1B-i9u5-Wa3tu2  
LV Write Access         read/write  
LV Creation host, time esprit-virtual-machine, 2022-11-04 14:58:13 +0100  
LV Status                available  
# open                  0  
LV Size                 1.00 GiB  
Current LE              256  
Segments                1  
Allocation              inherit  
Read ahead sectors      auto  
- currently set to      256  
Block device            253:0
```



# Créer des partitions : *gdisk*

```
Command (? for help): n
Partition number (1-128, default 1): 1
First sector (34-10485726, default = 2048) or {+-}size{KMGTP}:
Last sector (2048-10485726, default = 10485726) or {+-}size{KMGTP}: +500M
Current type is 8300 (Linux filesystem)
Hex code or GUID (L to show codes, Enter = 8300): ef00
Changed type of partition to 'EFI system partition'
```

*Créer une première partition de type « ef00 »*

```
Command (? for help): p
Disk /dev/sda: 10485760 sectors, 5.0 GiB
Model: VMware Virtual S
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): 58B696A1-E1FE-4A8F-8184-7421691F63AF
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 10485726
Partitions will be aligned on 2048-sector boundaries
Total free space is 9461693 sectors (4.5 GiB)

Number  Start (sector)    End (sector)  Size      Code  Name
   1            2048         1026047   500.0 MiB   EF00   EFI system partition
```

*Afficher la liste des partitions*

# Créer des partitions : *gdisk*

```
Command (? for help): i
Using 1
Partition GUID code: C12A7328-F81F-11D2-BA4B-00A0C93EC93B (EFI system partition)
Partition unique GUID: 201CE7C4-5D70-4AD5-8C44-5E91C4218151
First sector: 2048 (at 1024.0 KiB)
Last sector: 1026047 (at 501.0 MiB)
Partition size: 1024000 sectors (500.0 MiB)
Attribute flags: 0000000000000000
Partition name: 'EFI system partition'
```

*Afficher des informations  
relatives à une partition*

```
Command (? for help): c
Using 1
Enter name: /dev/sdb1

Command (? for help): p
Disk /dev/sdb: 10485760 sectors, 5.0 GiB
Model: VMware Virtual S
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): 5EA2CDAC-0BAA-492F-99B4-4D4E245D7462
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 10485726
Partitions will be aligned on 2048-sector boundaries
Total free space is 9461693 sectors (4.5 GiB)

Number  Start (sector)    End (sector)  Size      Code  Name
   1            2048         1026047   500.0 MiB   EF00  /dev/sdb1
```

*Renommer la partition et  
afficher la liste des partitions*



# Créer des partitions : *gdisk*

```
Command (? for help): n
Partition number (2-128, default 2):
First sector (34-10485726, default = 1026048) or {+}size{KMGTP}:
Last sector (1026048-10485726, default = 10485726) or {+}size{KMGTP}: +4G
Current type is 8300 (Linux filesystem)
Hex code or GUID (L to show codes, Enter = 8300): 8300
Changed type of partition to 'Linux filesystem'
```

*Créer une seconde partition  
de type « Linux filesystem »*

```
Command (? for help): c
Partition number (1-2): 2
Enter name: /dev/sdb2

Command (? for help): p
Disk /dev/sdb: 10485760 sectors, 5.0 GiB
Model: VMware Virtual S
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): 63651A19-35E1-47B3-879B-60F8E254D682
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 10485726
Partitions will be aligned on 2048-sector boundaries
Total free space is 1073085 sectors (524.0 MiB)
```

*Renommer la partition et  
afficher la liste des partitions*

Number	Start (sector)	End (sector)	Size	Code	Name
1	2048	1026047	500.0 MiB	EF00	/dev/sdb1
2	1026048	9414655	4.0 GiB	8300	/dev/sdb2



# Créer des partitions : *gdisk*

```
Command (? for help): w

Final checks complete. About to write GPT data. THIS WILL OVERWRITE EXISTING
PARTITIONS!!

Do you want to proceed? (Y/N): Y
OK; writing new GUID partition table (GPT) to /dev/sdb.
The operation has completed successfully.
```

*Sauvegarder la  
table de partitions  
et quitter*

```
esprit@esprit-virtual-machine:~$ sudo gdisk -l /dev/sdb
GPT fdisk (gdisk) version 1.0.8

Partition table scan:
  MBR: protective
  BSD: not present
  APM: not present
  GPT: present

Found valid GPT with protective MBR; using GPT.
Disk /dev/sdb: 10485760 sectors, 5.0 GiB
Model: VMware Virtual S
Sector size (logical/physical): 512/512 bytes
Disk identifier (GUID): 63651A19-35E1-47B3-879B-60F8E254D682
Partition table holds up to 128 entries
Main partition table begins at sector 2 and ends at sector 33
First usable sector is 34, last usable sector is 10485726
Partitions will be aligned on 2048-sector boundaries
Total free space is 1073085 sectors (524.0 MiB)
```

Number	Start (sector)	End (sector)	Size	Code	Name
1	2048	1026047	500.0 MiB	EF00	/dev/sdb1
2	1026048	9414655	4.0 GiB	8300	/dev/sdb2

*Réafficher la table de partitions*

# Créer des partitions : *parted*

```
esprit@esprit-virtual-machine:~$ sudo parted /dev/sdb
[sudo] password for esprit:
GNU Parted 3.4
Using /dev/sdb
Welcome to GNU Parted! Type 'help' to view a list of commands.
(parted) help
  align-check TYPE N          check partition N for TYPE(min|opt)
  alignment                   alignment
  help [COMMAND]              print general help, or help on
  COMMAND                     COMMAND
  mklabel,mktable LABEL-TYPE create a new disklabel (partition
  table)
  mkpart PART-TYPE [FS-TYPE] START END make a partition
  name NUMBER NAME            name partition NUMBER as NAME
  print [devices|free|list,all|NUMBER] display the partition table,
  available devices, free space, all found partitions, or a particular
  partition
  quit                        exit program
  rescue START END            rescue a lost partition near START
```

*Lister les options de  
parted*

```
(parted) print
Model: VMware, VMware Virtual S (scsi)
Disk /dev/sdb: 5369MB
Sector size (logical/physical): 512B/512B
Partition Table: gpt
Disk Flags:

Number  Start   End     Size    File system  Name        Flags
  1      1049kB  525MB   524MB    ext4          /dev/sdb1   boot, esp
  2      525MB   4820MB  4295MB    ext4          /dev/sdb2
```

*Afficher la liste des partitions*



# Créer des partitions : *parted*

```
(parted) mkpart  
Partition name? []? /dev/sdb3  
File system type? [ext2]? ext2  
Start?  
Start? 4820  
End? 5000
```

*Créer une nouvelle partition*

```
(parted) print  
Model: VMware, VMware Virtual S (scsi)  
Disk /dev/sdb: 5369MB  
Sector size (logical/physical): 512B/512B  
Partition Table: gpt  
Disk Flags:  
  
Number  Start  End      Size    File system  Name      Flags  
1       1049kB 525MB   524MB   /dev/sdb1    boot, esp  
2       525MB  4820MB  4295MB  /dev/sdb2  
3       4820MB 5000MB  179MB   /dev/sdb3
```

*Ré afficher la table des partitions*

```
Number  Start  End      Size    File system  Name      Flags  
1       1049kB 525MB   524MB   /dev/sdb1    boot, esp  
2       525MB  4820MB  4295MB  /dev/sdb2  
3       4820MB 5000MB  179MB   /dev/sdb3
```

*Quitter*

```
(parted) quit
```

---

# Créer un système de fichiers

---

- Pour créer un système de fichiers sur une partition, on utilise l'utilitaire *mkfs*
- *mkfs* est utilisé pour formater un système de fichiers Linux sur un périphérique, généralement une partition de disque dur.
- Les commandes suivantes sont équivalentes:
  - *mkfs.ext3 /dev/hda1*
  - *mkfs -t ext3 /dev/hda1*
- Pour créer un système de fichiers *étendu* sur une partition, on utilise l'utilitaire *mke2fs*

---

# Créer un système de fichiers

---

## *Créer un système de fichiers ext4*

```
esprit@esprit-virtual-machine:~$ sudo mkfs -t ext4 /dev/sdb2
mke2fs 1.46.5 (30-Dec-2021)
Creating filesystem with 1048576 4k blocks and 262144 inodes
Filesystem UUID: c45fe48a-499e-4692-9511-cd632f4c3500
Superblock backups stored on blocks:
    32768, 98304, 163840, 229376, 294912, 819200, 884736

Allocating group tables: done
Writing inode tables: done
Creating journal (16384 blocks): done
Writing superblocks and filesystem accounting information: done
```

## *Créer un système de fichiers FAT32*

```
esprit@esprit-virtual-machine:~$ sudo mkfs.fat -F 32 /dev/sdb1
mkfs.fat 4.2 (2021-01-31)
```



# Contrôler un système de fichiers

- Pour afficher des informations sur le système de fichiers d'une partition telles que type de système de fichiers, nombre d'inodes, date de création, date de dernier montage, date de dernière écriture, etc on utilise l'utilitaire *dumpe2fs*

```
esprit@esprit-virtual-machine:~$ sudo dumpe2fs /dev/sdb2
[sudo] password for esprit:
dumpe2fs 1.46.5 (30-Dec-2021)
Filesystem volume name:      <none>
Last mounted on:             <not available>
Filesystem UUID:             c45fe48a-499e-4692-9511-cd632f4c3500
Filesystem magic number:     0xEF53
Filesystem revision #:       1 (dynamic)
Filesystem features:         has_journal ext_attr resize_inode dir_index filetype e
xtent 64bit flex_bg sparse_super large_file huge_file dir_nlink extra_isize meta
data_csum
Filesystem flags:            signed_directory_hash
Default mount options:       user_xattr acl
Filesystem state:            clean
Errors behavior:             Continue
Filesystem OS type:          Linux
Inode count:                 262144
Block count:                 1048576
Reserved block count:        52428
Overhead clusters:           36936
Free blocks:                 1011634
Free inodes:                 262133
First block:                 0
```

*Afficher des informations sur le système de fichiers*

```
esprit@esprit-virtual-machine:~$ sudo dumpe2fs -b /dev/sdb2
dumpe2fs 1.46.5 (30-Dec-2021)
```

*Afficher les blocs défectueux du système de fichiers*

---

# Modifier un système de fichiers

---

- Pour modifier un système de fichiers **étendu**, on utilise l'utilitaire **tune2fs**
- Les principales options sont les suivantes:
  - **-l**: Liste le contenu du superbloc d'un système de fichiers
  - **-L**: Donne un label au volume d'un système de fichiers
  - **-j**: Permet de passer de ext2 à ext3

```
esprit@esprit-virtual-machine: ~$ sudo tune2fs -j /dev/sdb2  
tune2fs 1.46.5 (30-Dec-2021)  
The filesystem already has a journal.
```

# Modifier un système de fichiers

*Donner le label « Hello3A » au volume du système de fichiers sur la partition /dev/sdb2*

```
esprit@esprit-virtual-machine:~$ sudo tune2fs -L Hello3A /dev/sdb2  
tune2fs 1.46.5 (30-Dec-2021)
```

*Lister le contenu du superbloc*

```
esprit@esprit-virtual-machine:~$ sudo tune2fs -l /dev/sdb2  
tune2fs 1.46.5 (30-Dec-2021)  
Filesystem volume name: Hello3A  
Last mounted on: <not available>  
Filesystem UUID: c45fe48a-499e-4692-9511-cd632f4c3500  
Filesystem magic number: 0xEF53  
Filesystem revision #: 1 (dynamic)  
Filesystem features: has_journal ext_attr resize_inode dir_index filetype e  
xtent 64bit flex_bg sparse_super large_file huge_file dir_nlink extra_isize meta  
data_csum  
Filesystem flags: signed_directory_hash  
Default mount options: user_xattr acl  
Filesystem state: clean  
Errors behavior: Continue  
Filesystem OS type: Linux
```



---

# Vérifier un système de fichiers

---

- Si le système de fichier est endommagé ou corrompu, l'utilitaire *fsck* est utilisé pour vérifier et corriger le système
- L'option *-i* permet de corriger le fichier corrompu
- Remarque: pour des raisons de sécurité des données présentes sur le disque, il est fortement déconseillé de lancer un *fsck* sur une partition montée

```
esprit@esprit-virtual-machine: ~$ sudo fsck /dev/sdb2
fsck from util-linux 2.37.2
e2fsck 1.46.5 (30-Dec-2021)
Hello3A: clean, 11/262144 files, 36942/1048576 blocks
```

---

# Monter/Démonter un système de fichiers d'une manière temporaire

---

- Pour monter un système de fichiers, on utilise l'utilitaire *mount*
- Les principales options sont les suivantes :
  - *-a*: pour monter tous les systèmes de fichiers déclarés dans le fichier */etc/fstab*
  - *-t*: pour préciser le type du système de fichiers
  - *-o*: pour préciser les options de montage (exemple l'option *-o* suivie de *-ro* pour monter un système de fichiers en lecture seule)
- Pour démonter un système de fichiers, on utilise l'utilitaire *umount*

# Monter/Démonter un système de fichiers d'une manière temporaire

*Monter la partition /dev/sdb2 sur le répertoire /3A*

```
esprit@esprit-virtual-machine:~$ sudo mkdir /3A
esprit@esprit-virtual-machine:~$ sudo mount /dev/sdb2 /3A
esprit@esprit-virtual-machine:~$ df -h
```

Filesystem	Size	Used	Avail	Use%	Mounted on
tmpfs	389M	2.0M	387M	1%	/run
/dev/sda3	20G	13G	5.3G	71%	/
tmpfs	1.9G	0	1.9G	0%	/dev/shm
tmpfs	5.0M	4.0K	5.0M	1%	/run/lock
/dev/sda2	512M	5.3M	507M	2%	/boot/efi
tmpfs	389M	3.5M	386M	1%	/run/user/1000
/dev/sdb2	3.9G	24K	3.7G	1%	/3A

*Utiliser la commande **df** pour afficher des statistiques sur les systèmes de fichiers montés*

```
esprit@esprit-virtual-machine:~$ cd /3A
esprit@esprit-virtual-machine:/3A$ df .
```

Filesystem	1K-blocks	Used	Available	Use%	Mounted on
/dev/sdb2	4046560	24	3820440	1%	/3A



# Montage permanent d'un système de fichiers

- Le fichier */etc/fstab* contient tous les fichiers systèmes et les informations relatives qui seront utilisées pour monter un lecteur au démarrage du système

*Exemple du fichier /etc/fstab*

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda3 during installation
UUID=e4d6ddf0-dee0-4c17-aa79-b8a986bd2b1f / ext4 errors=remount-ro 0 1
# /boot/efi was on /dev/sda2 during installation
UUID=C2A5-8331 /boot/efi vfat umask=0077 0 1
/swapfile none swap sw 0 0
/dev/sdb2 /3A ext4 defaults 0 0
```

1

2

3

4

5

6

- 1: Emplacement physique du système de fichiers
- 2: Le point de montage (le répertoire doit exister)
- 3: Le type du systèmes de fichiers
- 4: Les options de montage (séparées par des virgules)
- 5: Sauvegarde via l'utilitaire dump (1=sauvegarder la partition)
- 6: Ordre de vérification (0=ne pas vérifier)

---

# Options de montage

---

- **ro** pour monter le système de fichiers en lecture seule,
- **rw** pour monter le système de fichiers en lecture-écriture,
- **noauto** pour que le système de fichiers ne soit pas monté au démarrage (l'option contraire est : **auto**),
- **user** pour qu'un simple utilisateur puisse monter et démonter le système de fichiers et pas seulement le root (l'option contraire : **nouser**),
- **uid**, **gid** et **umask** pour définir des permissions pour l'ensemble du système de fichiers (pour les systèmes déficients comme FAT ou NTFS),
- **defaults** pour les options par défaut (notamment **rw**, **auto** et **nouser**),
- **sw** pour les systèmes de swap.

---

# Quota

---

---

# Types de Quota

---

- Les quotas permettent de poser des limites à l'utilisation du système de fichiers.
- Il existe deux limites :
  1. Limite sur le nombre d'inodes donc sur le nombre de fichiers créés
  2. Limite sur le nombre de blocs donc sur la taille du disque ou espace utilisé
- Pour chaque limite, il existe deux types de limites :
  1. La limite matérielle (dure ou hard): spécifie la limite absolue, sur le disque géré par des quotas, au delà de laquelle l'utilisateur ne peut pas aller.
  2. La limite logicielle (souple ou soft): l'utilisateur peut dépasser cette limite mais dans ce cas le système affiche un warning

# Mise en place des quotas

Installer le paquetage *quota* s'il n'existe pas

```
esprit@esprit-virtual-machine:~$ sudo apt-get install quota
```

Ajouter les mots clés *usrquota* et/ou *grpquota* dans le fichier /etc/fstab dans les options de montage de la partition en question

```
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point> <type> <options> <dump> <pass>
# / was on /dev/sda3 during installation
UUID=e4d6ddf0-dee0-4c17-aa79-b8a986bd2b1f / ext4 errors=remount-ro 0 1
# /boot/efi was on /dev/sda2 during installation
UUID=C2A5-8331 /boot/efi vfat umask=0077 0 1
/swapfile none swap sw 0 0
/dev/sdb2 /3A ext4 defaults,usrquota,grpquota 0 0
```

Remonter les systèmes de fichiers

```
esprit@esprit-virtual-machine:~$ sudo mount -a
```



---

# Mise en place des quotas

---

Vérifier et mettre jour la configuration

```
esprit@esprit-virtual-machine:~$ sudo quotacheck -aucvqm
quotacheck: Your kernel probably supports ext4 quota feature but you are using external quota files.
Please switch your filesystem to use ext4 quota feature as external quota files on ext4 are deprecated.
quotacheck: Scanning /dev/sdb2 [/3A] done
quotacheck: Cannot stat old user quota file /3A/aquota.user: No such file or directory. Usage will not be subtracted.
quotacheck: Cannot stat old group quota file /3A/aquota.group: No such file or directory. Usage will not be subtracted.
quotacheck: Cannot stat old user quota file /3A/aquota.user: No such file or directory. Usage will not be subtracted.
quotacheck: Cannot stat old group quota file /3A/aquota.group: No such file or directory. Usage will not be subtracted.
quotacheck: Checked 3 directories and 0 files
quotacheck: Old file not found.
quotacheck: Old file not found.
```

Mettre en place les quotas disque

```
esprit@esprit-virtual-machine:~$ sudo quotaon -av
quotaon: Your kernel probably supports ext4 quota feature but you are using external quota files. Please switch your filesystem to use ext4 quota feature as external quota files on ext4 are deprecated.
./dev/sdb2 [/3A]: group quotas turned on
./dev/sdb2 [/3A]: user quotas turned on
```

# Mise en place des quotas

Configurer les limites soft et hard:

- Par utilisateur: *edquota -u user*
- Par groupe: *edquota -g group*

```
esprit@esprit-virtual-machine: $ sudo edquota -u user1
```



```
Disk quotas for user user1 (uid 1004):
```

Filesystem	blocks	soft	hard	inodes	soft	hard
/dev/sdb2	0	0	0	0	0	0



```
Disk quotas for user user1 (uid 1004):
```

Filesystem	blocks	soft	hard	inodes	soft	hard
/dev/sdb2	0	0	0	0	3	5

# Mise en place des quotas

```
root@esprit-virtual-machine:/home/esprit# su user1
$ cd /3A
$ touch file1
$ touch file2
$ touch file3
$ touch file4
$ touch file5
$ touch file6
touch: cannot touch 'file6': Disk quota exceeded
```

Pour lister les quotas utilisateurs ou groupes, on utilise la commande *quota*

```
$ quota
Disk quotas for user user1 (uid 1004):
  Filesystem  blocks    quota   limit   grace  files   quota   limit   grace
   /dev/sdb2         0         0       0         5*        3        5    6days
```



# Rapport de quota

Pour afficher un rapport de quota, on utilise la commande *repquota* avec les options:

- *-a*: Rapport sur tous les systèmes de fichiers indiquées dans */etc/mtab* en lecture-écriture avec quotas
- *-g*: Rapport pour le groupe

```
esprit@esprit-virtual-machine: ~$ sudo repquota -a
*** Report for user quotas on device /dev/sdb2
Block grace time: 7days; Inode grace time: 7days

```

		Block limits				File limits			
User		used	soft	hard	grace	used	soft	hard	grace
root	--	20	0	0		2	0	0	
test	++	0	0	0		3	2	3	6days
user1	++	0	0	0		5	3	5	6days