

# TD 4

## Gestion des ressources mémoire

# Gestion de la mémoire

Les différentes mémoires:

- La mémoire **vive** “RAM” : sa particularité est qu’elle est très rapide (bande passante max : 25Go/s en DDR4) et volatile.
- La mémoire **morte** “ROM” (Disques durs): à l’inverse de la première, celle-ci est persistante après l’arrêt du PC mais elle est comparativement à la RAM très lente (maximum 600Mo/s en SAS 2.0, et généralement 100Mo/s en RAID).
- La mémoire **cache**: exemple du microprocesseur ou du disque dur qui disposent de cache interne (architecture) et externe (dans la RAM). Le cache garde en mémoire des données auxquelles le processeur ou le disque accèdent de manière répétitive. Objectif: accélérer les échanges avec les périphériques « lents » (par rapport à la RAM). Ordre de grandeur (cache interne) pour les disques durs : ~8-64 Mo/dd

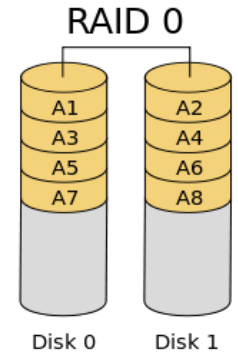
# Disques Durs

- Un disque dur est composé de plusieurs disques ou plateaux magnétiques.
- Sur chacun de ces disques une tête de lecture/écriture se déplace en ligne droite.
- La performance d'un disque dépendra de la vitesse de rotation (jusqu'à 30000rpm), du nombre de plateaux, du temps d'accès (déplacement de la tête de lecture), du débit (lié au contrôleur).
- Contrôleurs :
  - Anciens : ISCSI (1979) IDE (1991)
  - Actuels : SATA (2003), SAS (2009)
- Disques SSD : Pas de mécanique, il s'agit globalement d'une série de mémoires flash. Beaucoup plus performants en terme de lecture/écriture, mais coût/mo élevé et nombre d'écritures limité (dégradation)
- Coût du stockage en baisse exponentielle : \$180/5Mo en 1980 ➔ \$15/To en 2021

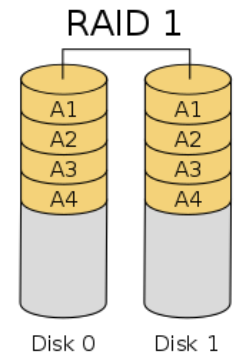
# Gestion de la mémoire

Les différents RAID (les principaux):

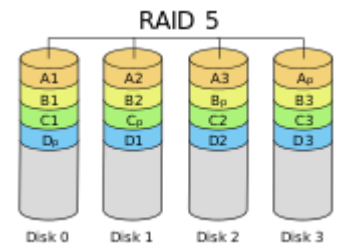
RAID 0 : également connu sous le nom d'« entrelacement de disques » ou de « volume agrégé par bandes » (*striping* en anglais), permet d'augmenter significativement les performances de la grappe de dd en faisant travailler n disques durs en parallèle (avec  $n \geq 2$ ) (NB: 1dd HS = tout est perdu).



RAID 1 : consiste en l'utilisation de n disques redondants (avec  $n \geq 2$ ), chaque disque de la grappe contenant à tout moment exactement les mêmes données, d'où l'utilisation du mot « miroir ».



RAID 5 : La parité, qui est incluse avec chaque écriture se retrouve répartie circulairement sur les différents disques. Chaque bande est donc constituée de N-1 blocs de données (N = nb de dd) et d'un bloc de parité. En cas de défaillance de l'un des disques, pour chaque bande il manquera soit un bloc de données soit le bloc de parité. Si c'est le bloc de parité, ce n'est pas grave, car aucune donnée ne manque. Si c'est un bloc de données, on peut calculer son contenu à partir des  $N - 2$  autres blocs de données et du bloc de parité.



# Gestion de la mémoire vive

Le noyau de Linux met en place des systèmes complexes pour gérer au mieux la mémoire dont il dispose et les besoins des programmes en mémoire vive.

Nous allons nous intéresser d'un façon globale à la gestion des ressources de mémoire.

# La mémoire virtuelle

La mémoire vive étant chère, Linux supporte de la mémoire virtuelle, c'est à dire que l'on utilise de l'espace de stockage du disque dur pour étendre la mémoire vive quand cela est nécessaire. La mémoire virtuelle est aussi appelée **swap**.

L'utilisation du swap doit être exceptionnelle, elle permet au système de continuer à fonctionner mais en dégradant les performances.

Si un système swap beaucoup ... il faut ajouter de la RAM

Sur un système peu chargé

Mémoire vive (RAM)		Mémoire virtuelle (swap)
noyau	processus	Cache disque

Sur un système chargé :

Mémoire vive (RAM)		Mémoire virtuelle (swap)	
noyau	processus	Cache disque	processus

# Le swap (l'espace d'échange)

Linux peut utiliser soit un fichier normal (fichier d'échange - `pagefile.sys` sous W10), soit une partition dédiée pour le swap (espace d'échange).

Un fichier d'échange peut être créé manuellement sous Linux (`file.swap` – 512 Mo):

```
#sudo fallocate -l 512m /file.swap  
#sudo chmod 600 /file.swap  
#sudo mkswap /file.swap  
#sudo swapon /file.swap  
#vi /etc/fstab - - - > /file.swap none swap defaults 0 0
```

L'utilisation d'une partition est plus efficace car le noyau n'a pas à gérer le système de fichier et une partition de swap a un formatage adapté ( $\neq$  de celui des autres partitions).

Le swap n'est pas obligatoire si on privilégie la performance et qu'on veut être sûr de ne pas avoir de mode dégradé.

Le swap est activable ou désactivable à la volée avec la commande :

```
#swapon -a // pour activer tous les volumes d'échange  
#swapoff -a // pour désactiver tous les volumes d'échange
```

# Terminologie

- Disques chiffants : chiffrement matériel
- Virtualisation / Conteneurisation : Mutualisation et partages des ressources physiques (dont mémoire)
- NAS (Network Attached Storage) vs SAN (Storage Area Network)
- Déduplication : Technique de stockage de données, consistant à factoriser des séquences de données identiques afin d'économiser l'espace utilisé



# Exercices

## 1- Swap

- Vérifiez l'état de la mémoire avant et après activation et désactivation du swap
- Utilisez la commande **free** pour visualiser la mémoire

## 2- Les différents espaces de mémoire

- À l'aide de la commande interactive **top**, listez les processus les plus consommateurs de mémoire
- Expliquez les paramètres Virt, Res, Srh

# Exercices

## 3- Impact des caches

- Par simplicité passez en root : ***sudo -i***
- Avec la commande **free** notez la valeur des caches
- purgez les caches : ***echo 3 > /proc/sys/vm/drop\_caches***
- Avec **free**, validez que l'espace alloué aux caches a diminué
- Lancer une commande avec beaucoup d'accès disque: la commande est lancée avec **time** pour mesurer sa durée d'exécution:

*Par exemple: **time find / -type f -name '\*.conf'***

- Vérifier l'évolution des caches avec **free**
- On relance la même commande:
  1. Que constatez-vous sur la durée de l'opération
  2. Pour valider : purgez les caches et relancez la commande
  3. Expliquez les résultats