

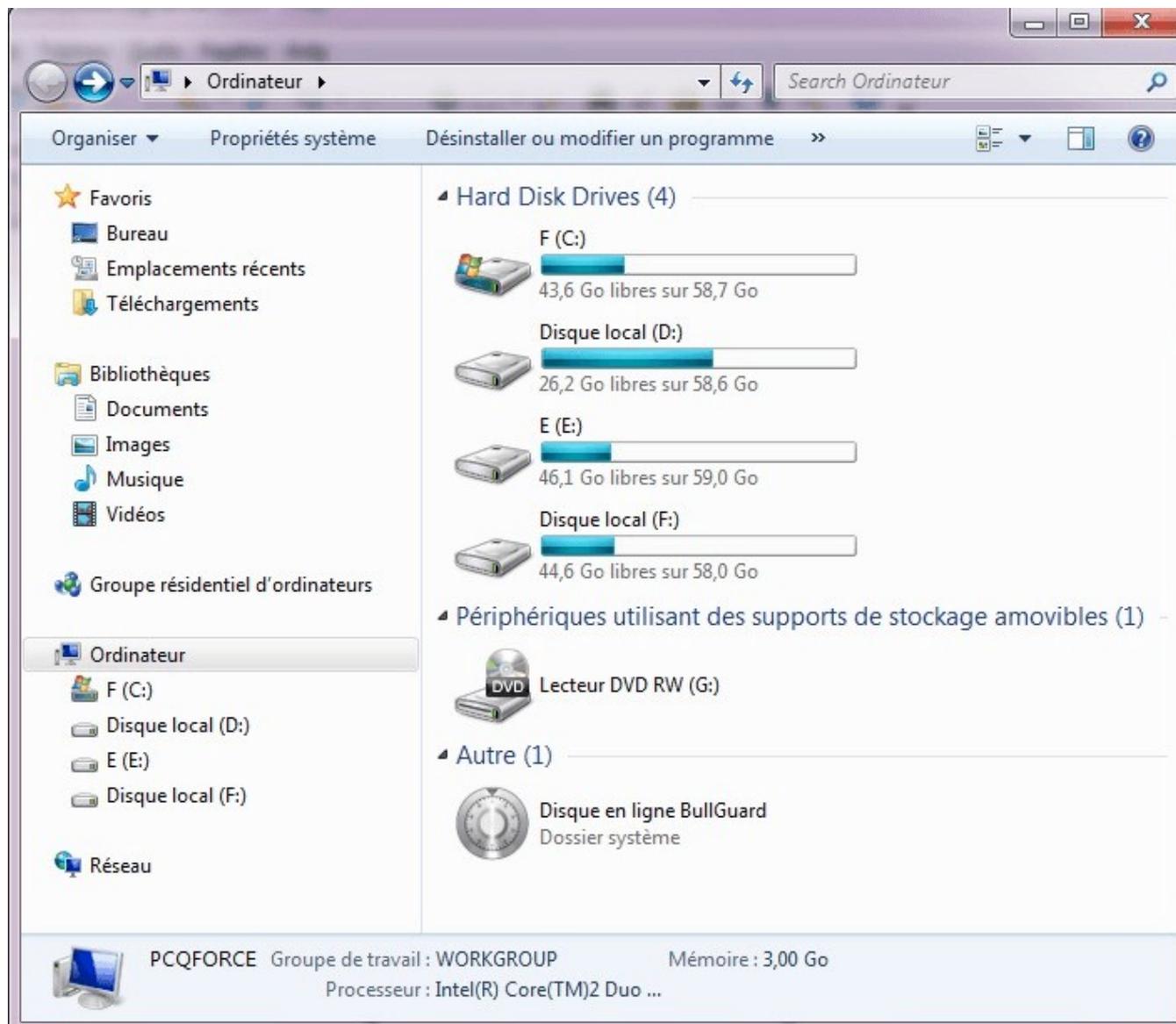
Organiser les fichiers

Aspects pratiques,
Partition des disques,
Systèmes de fichiers

Aujourd'hui

- Un cours qui va des bases à des concepts plus avancés
- Un cours où l'on utilise la ligne de commande sous Windows puis sous Linux (on apprend au passage ce qu'est une machine virtuelle)
- Programme :
 - Aspects pratiques : créer fichiers et répertoires sous Windows et Linux, en ligne de commande
 - Découpage des disques en partition
 - Formatage des partitions en systèmes de fichiers

Organisation des fichiers sous Windows



Organisation des fichiers sous Windows

- Plusieurs **lecteurs** identifiés par une lettre : C :, D :, ...
- Autrefois A : voire B : pour les disquettes
- Souvent : les **lecteurs réseau** ont les dernières lettres de l'alphabet
- Un lecteur contient des **dossiers** qui contiennent eux-mêmes des **dossiers** ou des **fichiers**
- C : ne peut contenir qu'une fois le dossier « X » à la racine, par contre D : peut également le contenir

Organisation des fichiers sous Windows

- En général, C : est le lecteur principal, qui contient l'installation de Windows et les programmes (en général)
- Windows est contenu dans **C:\Windows**
- Les programmes sont dans **C:\Program Files** (x86) et **C:\Program Files** qui apparaît souvent dans la version française comme **C:\Programmes** (pourquoi ces distinctions?)
- Les données utilisateurs sont dans **C:\Users** nommé **C:\Utilisateurs** dans la version française

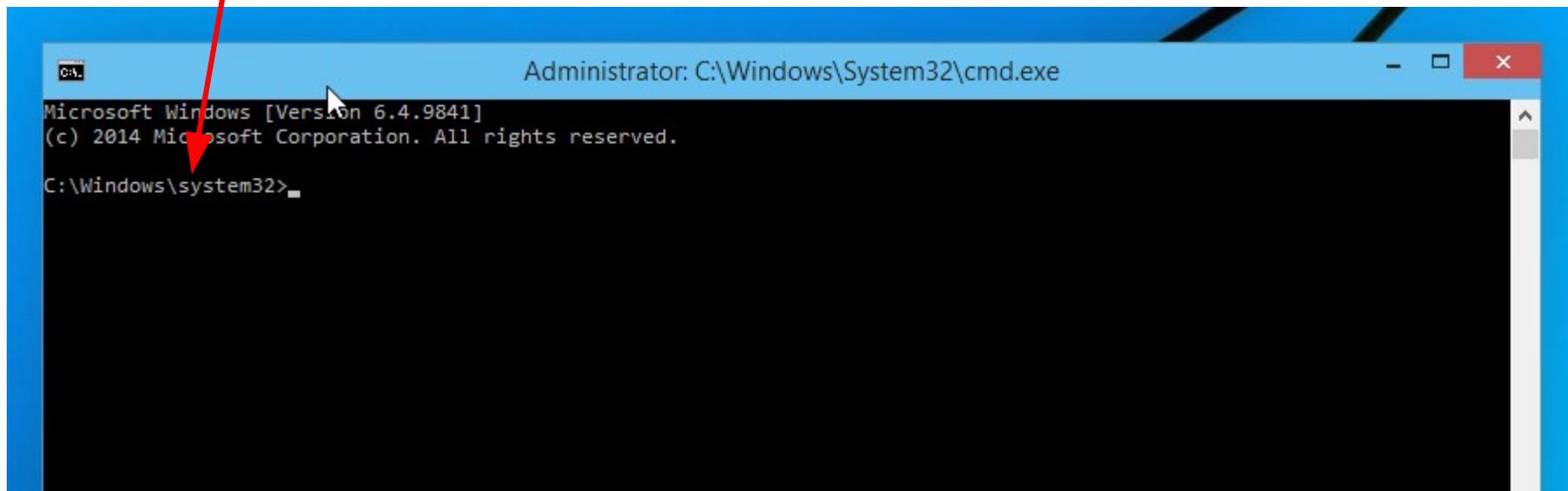
Organisation des fichiers sous Windows

- On peut parcourir fichiers et dossiers dans l'Explorateur Windows
- Pour créer un dossier (et parfois un fichier) : clic-droit → Nouveau → dossier puis entrer le nom
- On va apprendre à utiliser la **ligne de commande**, qui vous sera utile notamment pour la programmation

Invite de commandes Windows

- Accessible par :
Clic-gauche sur la petite loupe à côté du menu Démarrer, puis taper « cmd » et taper sur Entrée

Répertoire courant



The screenshot shows a Windows Command Prompt window titled "Administrator: C:\Windows\System32\cmd.exe". The window content displays the following text:

```
Microsoft Windows [Version 6.4.9841]  
(c) 2014 Microsoft Corporation. All rights reserved.  
C:\Windows\system32>
```

A red arrow points from the text "Répertoire courant" to the path "C:\Windows\system32" in the command prompt.

Changer de répertoire courant

- **Gymnastique très importante** : aller dans le bon répertoire
- On est dans le répertoire qui précède le signe > (invite de commande)
- Plusieurs options :
 - On est dans le mauvais lecteur, par exemple C :, et on veut aller dans D :. On tape alors D :
 - On veut aller dans un répertoire que l'on connaît, par exemple **C:\Windows**. On tape « cd C:\Windows ». On parle de **chemin absolu**
 - **Chemin relatif** : cf slide suivante

Changer de répertoire courant

- Chemin relatif : permet de ne pas taper tout le chemin d'accès à un répertoire, mais de le situer par rapport au répertoire courant
- « cd .. » ramène au répertoire parent dans l'arborescence
- « cd . » ne fait rien, car le point représente le répertoire courant
- « cd ABC » va dans le sous-répertoire ABC du répertoire courant
- Pour voir les sous-répertoires (et les fichiers) contenus dans le répertoire courant, taper « dir »
- Un « truc » utile : l'autocomplétion. Commencez à taper un nom, et appuyez sur la touche Tabulation pour voir...

Créer un dossier, un fichier...

- Créer un sous-dossier ABC du répertoire courant : « `mkdir ABC` »
- Créer un sous-dossier DEF de ABC :
 - Aller dans ABC : « `cd ABC` »
 - Puis « `mkdir DEF` »
- Retourner dans ABC depuis DEF ? « `cd ..` »

Créer un dossier, un fichier...

- Les commandes renvoient (parfois) des données. Exemple : **dir**.
- On peut **rediriger** ces données vers un fichier :
 - « `dir > liste_fichiers.txt` » écrit la liste des fichiers dans `liste_fichiers.txt` (en écrasant tout ce que contenait ce fichier s'il existait)
 - « `dir >> liste_fichiers.txt` » écrit la liste des fichiers **à la fin** de `liste_fichiers.txt` (en n'écrasant donc rien de ce que contenait ce fichier s'il existait)
- On peut ouvrir `liste_fichiers.txt`, par exemple en faisant « `notepad liste_fichiers.txt` »

Créer un dossier, un fichier...

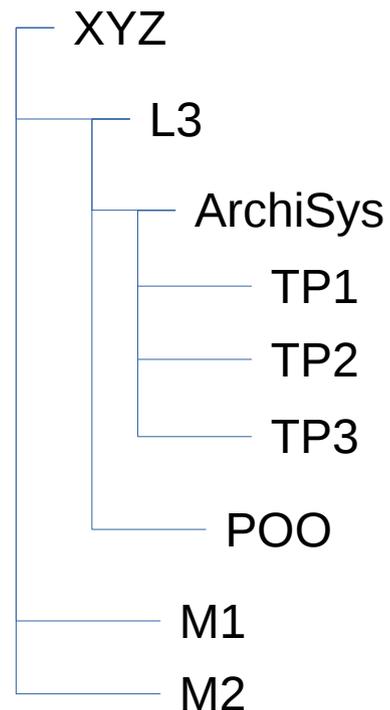
- Les commandes renvoient (parfois) des données. Exemple : echo.
- « echo Bonjour » renvoie Bonjour.
- Question : comment écrire Bonjour dans hello.txt ?
- Question 2 : comment rajouter Guten Tag à la fin de hello.txt ?

Créer un dossier, un fichier...

- Windows ne différencie pas les noms avec ou sans majuscule.
- Unix / Linux le fait.
- Que fait :
 - « `dir > liste_fichiers.txt` »
 - « `echo LOUPE > Liste_Fichiers.txt` »
 - « `notepad liste_fichiers.txt` » ???

TP partie 1 : invite de commande Windows

- Ouvrir l'invite de commande Windows
- Dans votre répertoire personnel (C:\Users\XYZ), créez l'arborescence suivante :



TP partie 1 : invite de commande Windows

- Dans XYZ\L3\ArchiSys\TP1, créez un fichier « liste_fichiers_XYZ.txt » contenant le contenu du répertoire utilisateur XYZ
- Ajoutez à la fin de ce fichier « Ainsi ce termine la liste des fichiers de mon répertoire utilisateur. »
- On peut copier un fichier avec la commande « copy ». Par exemple, « copy A.txt B\C.txt » copie le fichier A.txt sous le nouveau nom C.txt à l'intérieur du dossier B.
- Copier XYZ\L3\ArchiSys\TP1\liste_fichiers_XYZ.txt sous le nom liste_fichiers.txt dans le répertoire XYZ.

TP partie 1 : invite de commande Windows

- On peut déplacer un fichier à l'aide de la commande « move », qui s'utilise comme la commande « copy ».
 - Créer un fichier XYZ\hello.txt contenant le texte « Bonjour »
 - Déplacer ce fichier dans XYZ\L3\ArchiSys\TP1\
- Aller dans XYZ\L3\ArchiSys\TP2\ en utilisant des chemins relatifs et créer un fichier « prevision.txt » contenant « Ceci devrait être un TP d'assembleur ».
- Aller dans XYZ\L3\ArchiSys\TP3\ en utilisant des chemins relatifs et créer un fichier « prevision.txt » contenant « Ceci devrait être un TP plus avancé sur Linux ».

TP partie 1 : invite de commande Windows

- Dans XYZ\L3\ArchiSys\TP1\, créez un fichier « erreur.txt » contenant « J'ai été crée par erreur ! »
- Déplacez ce fichier vers le dossier parent.
- Supprimez ce fichier à l'aide de la commande « del ».

TP partie 1 : invite de commande Windows

- Dans XYZ\L3\ArchiSys\TP1\, créez un dossier Erreur
- Dans ce dossier Erreur, créez un fichier erreur.txt contenant « Erreur »
- Essayez de copier le dossier Erreur sous le nouveau nom Erreur2. Que se passe-t-il ?
- Utilisez maintenant la commande « xcopy » pour créer Erreur3. Vous obtiendrez une « bonne » copie du répertoire Erreur.

TP partie 1 : invite de commande Windows

- Essayez de supprimer les répertoire Erreur et Erreur3 à l'aide de la commande « rmdir » qui sert à effacer les répertoires. Que se passe-t-il ?
- Tapez « rmdir /? » qui permet d'avoir l'aide de la commande rmdir, et trouvez une solution au problème.
- Supprimez les dossier Erreur et Erreur3 et tout leur contenu grâce à cette solution. Supprimez aussi Erreur2.

TP partie 2 : invite de commande Linux

- Linux est un autre **systeme d'exploitation**, alternatif à Windows, dont nous parlerons plus tard dans le cours
- On peut l'installer à la place de Windows ou « en plus » (système de dual boot : on choisit au démarrage de l'ordinateur quel système d'exploitation utiliser)
- Sur ces machines, encore une autre solution : **machine virtuelle**. Ceci signifie qu'un logiciel va exécuter Linux dans Windows, dans une fenêtre, en « faisant croire » à Linux qu'il tourne sur un ordinateur virtuel.
- Solution parfois utile, mais attention : **performance moindre**.

TP partie 2 : invite de commande Linux

- **Wikipedia** : En informatique, une machine virtuelle (anglais virtual machine, abr. VM) est une illusion d'un appareil informatique créée par un logiciel d'émulation. Le logiciel d'émulation simule la présence de ressources matérielles et logicielles telles que la mémoire, le processeur, le disque dur, voire le système d'exploitation et les pilotes, permettant d'exécuter des programmes dans les mêmes conditions que celles de la machine simulée.

Un des intérêts des machines virtuelles est de pouvoir s'abstraire des caractéristiques de la machine physique utilisée (matérielles et logicielles — notamment système d'exploitation), permettant une forte portabilité des logiciels et la gestion de systèmes hérités étant parfois conçus pour des machines ou des environnements logiciels anciens et plus disponibles.

Les machines virtuelles sont également utilisées pour isoler des applications pour des raisons de sécurité, pour augmenter la robustesse d'un serveur en limitant l'impact des erreurs système ou pour émuler plusieurs machines sur une seule machine physique (virtualisation).

Les inconvénients des machines virtuelles sont d'une part des performances brutes sensiblement inférieures à une exécution sur le matériel en natif (ce qui peut rendre malaisé le test de jeux 3D), d'autre part de consommer une petite partie de la mémoire réelle pour leur propre fonctionnement. Leurs avantages sont de permettre des tests de variantes d'installation sur des machines simulées possédant de tailles de RAM, de mémoire graphique, et des nombres de processeurs très divers.

TP partie 2 : invite de commande Linux

- Pour lancer la VM : Démarrer → Oracle VM VirtualBox
→ Oracle VM VirtualBox
- Dans la liste des VM qui apparaît (il devrait y en avoir 3), sélectionner « Ubuntu 14.04.5 LTS (Miage) »
- Observez les caractéristiques de la VM : 2048 Mo de RAM lui sont attribués, 16 Mo de mémoire vidéo,... ce sont les paramètres de l'ordinateur virtuel que va simuler le logiciel.
- Appuyer sur le bouton Démarrer. Il faut ensuite attendre quelques instants.

TP partie 2 : invite de commande Linux

- Dans la barre d'outils à gauche, cliquer sur « Fichiers » (3^e icône en partant du haut).
- Dans la fenêtre qui s'ouvre, qui est dans votre dossier personnel, cliquez sur « Ordinateur » dans la partie de gauche.
- Que voyez-vous ?

TP partie 2 : invite de commande Linux

- Il n'y a plus de lecteurs C :, D :, etc. On y reviendra plus tard. A la place, tout est rangé à la **racine** notée « / ».
- Attention : on sépare les sous-dossiers avec « \ » sous Windows et avec « / » sous Linux.

TP partie 2 : invite de commande Linux

- Dans « / », on trouve par exemple :
 - /bin : on y range les applications (*binaries*) essentielles
 - /boot : on y range ce qui sert au démarrage de Linux
 - /dev : dossier dont les éléments permettent l'accès aux périphériques
 - /etc : fichiers de configuration
 - /home : contient les données des utilisateurs
 - /lib : contient les librairies essentielles (fonctionnalités partagées entre programmes)
 - /media : point d'accès aux autres lecteurs que le principal
 - /proc : informations sur les processus en cours d'exécution
 - /root : données du superutilisateur
 - /sbin : applications essentielles réservées au superutilisateur
 - /sys : informations système
 - /tmp : fichiers temporaires
 - /usr : logiciels, librairies, ... « moins importants »
 - /var : divers fichiers

TP partie 2 : invite de commande Linux

- Pour ouvrir une ligne de commande : cliquer sur la barre d'outils à gauche, sur la première icône en partant du haut, puis taper « terminal » et cliquer sur l'icône correspondante
- Une fenêtre s'ouvre, l'invite de commande est de la forme :

```
ubuntu@ubuntu:~$
```

- Ubuntu (le premier) : nom de l'utilisateur
- Ubuntu (le second) : nom de la machine
- ~ : identifiant de votre répertoire personnel (/home/ubuntu puisque l'utilisateur se nomme « ubuntu »)
- \$ sépare ces informations des commandes que vous allez taper

TP partie 2 : invite de commande Linux

- Pour avoir de l'aide sur une commande : taper « man commande »
- Testez : « man cp ». Que fait donc la commande « cp » ? A quoi correspond-elle dans l'invite de commandes Windows ?
- Refaire les exercices de la première partie du TP sous Linux, en utilisant les commandes correspondantes.
- Pour cela, on utilisera notamment « cd », « cp », « mv », « mkdir », « ls », « echo »
- Plutôt que « notepad », on peut utiliser « cat » pour lire un fichier. Les opérateurs « > » et « >> » fonctionnent comme sous l'invite de commande Windows.
- Pour rmdir et xcopy : chercher les bonnes options de « rm » et de « cp » dans le manuel.

TP partie 2 : invite de commande Linux

- Retourner dans le répertoire utilisateur (commande « `cd ~` ») puis afficher les fichiers et dossiers avec la commande « `ls` »
- Faire maintenant « `ls -l` ». Quelles sont ces informations supplémentaires ? On va en parler bientôt : il s'agit du propriétaire des fichiers et des permissions (qui peut les lire, les modifier, est-ce que le fichier peut être utilisé comme une application...)

TP partie 2 : invite de commande Linux

- Allez dans le répertoire `/usr/bin`.
- Affichez la liste des fichiers avec « `ls` ».
- Il y en a tellement qu'on ne peut pas tout lire...
une solution : faites « `ls | less` ». Appuyez sur
« `q` » pour quitter ce mode de lecture.
L'opérateur « `|` » **redirige** la sortie de `ls` vers
l'entrée de `less`.

TP partie 2 : invite de commande Linux

- On peut aussi afficher moins de fichiers, en spécifiant quelle forme doivent avoir les noms.
- En particulier : « g* » désigne tout fichier commençant par un g. « gp? » désigne tout fichier commençant par gp et suivi d'un seul caractère.
- Faites « ls g* » et « ls gp? » et observez les résultats de la commande. On peut les combiner aux options de la commande, en faisant par exemple « ls -l g* »
- Dans ce dernier cas, on observe que les fichiers appartiennent au superutilisateur (appelé root)

TP partie 3 : Windows

- On va installer « à la main » un utilitaire de manipulation d'images en ligne de commande, sous Windows puis sous Linux
- Allez télécharger sur www.imagemagick.org le fichier « ImageMagick-7.0.7-1-portable-Q16-x64.zip »
- Ce fichier finit par .zip, ce qui signifie qu'il est compressé
- Une fois le téléchargement terminé, décompressez le fichier (trouvez-le dans l'explorateur Windows puis clic-droit → 7-Zip → Extraire ici)
- Renommez le dossier résultant en « ImageMagick » et déplacez-le dans votre répertoire Documents

TP partie 3 : Windows

- Une commande essentielle d'ImageMagick est la commande « convert ». Tapez « convert » en ligne de commande.
- Ce n'est pas la commande qu'on veut ! Windows ne cherche pas au bon endroit... il va falloir mettre à jour la **variable d'environnement PATH** pour dire à Windows où il doit chercher les programmes dont on lui donne le nom.
- On fait ça pour ImageMagick, mais vous en aurez besoin pour installer gcc sur votre machine.

TP partie 3 : Windows

- Cliquez sur la petite loupe à côté de Démarrer et entrez « Panneau de configuration »
- Allez dans Système et Sécurité puis Système puis Paramètres Système Avancés puis Variables d'environnement
- Modifiez la variable PATH en ajoutant « ; C:\Users\VotreNomDUtilisateur\Documents\ImageMagick »
- Maintenant, tapez « convert » en ligne de commande : la sortie est différente car Windows utilise un autre logiciel quand on appelle convert, celui donné par ImageMagick (et plus celui de base de Windows)

TP partie 3 : Windows

- Allez sur www.imagemagick.org/Usage/text
- Amusez-vous à créer une image à partir d'un texte
- Si vous êtes très en avance sur le reste des étudiants, vous pouvez continuer à regarder le manuel en ligne. Essayez de trouver comment « convert » permet de redimensionner une photo.
- A la fin du cours, si le temps le permet, on apprendra à redimensionner toutes les photos d'un répertoire en une seule commande !

TP partie 4 : Linux

- Retournez sur la machine virtuelle Linux. ImageMagick n'est pas installé...
- L'installation sous Ubuntu est **très simple**.
- Faites « `sudo apt-get install imagemagick` » dans un terminal
- Vous pouvez maintenant utiliser exactement les mêmes commandes, dont « `convert` » !
- Installation d'un paquet sous Ubuntu : `sudo apt-get install nom_du_paquet`

TP partie 4 : Linux

- Pas besoin de changer le PATH car la structuration est différente
- Faites « whereis convert » : on voit que le fichier de l'application est « /usr/bin/convert ».
- En fait, sous Windows, les programmes sont rangés à un endroit qui leur est propre. Sous Linux, les fichiers du programme sont « éclatés » : les binaires (fichiers d'application) dans les dossiers tels que /usr/bin, les librairies dans /usr/lib, et ainsi de suite.
- Donc Linux va chercher dans /usr/bin et trouve directement le fichier qu'il veut.

Retour à la théorie !

Si vous avez fini très en avance, vous pouvez :

- Continuer à explorer les possibilités d'ImageMagick
- Chercher comment redimensionner un lot d'images avec une simple commande (sous Linux, c'est plus simple)
- Vous renseigner sur « apt-get » (et en parler avec moi), afin d'installer d'autres paquets.

Pourquoi ce .txt ?

- Notion d'**extension**. Format d'un fichier :
nom.extension
- Autrefois, sous DOS, fichier de 8 caractères (sans accents ou autres) suivi d'une extension de 3 caractères
- Aujourd'hui (ou sous UNIX), beaucoup plus libéral, mais **attention à ne pas mettre d'accents dans les noms de fichiers**

Pourquoi ce .txt ?

- **L'extension :**

- Indique le type du fichier (image, texte,...) mais aussi son encodage
- Un fichier n'est a priori qu'une suite de 0 et de 1 référencée par le système de fichiers : cette suite est interprétée par une application donnée
- Sous Windows, l'extension (ainsi que les préférences associées) dit quelle application ouvrira quel fichier
- Sous Unix, le format du fichier est contenu ailleurs et n'est pas dépendant de l'extension. On peut d'ailleurs avoir des fichiers sans extension. La commande « file » permet de connaître le type d'un fichier.

Quelles extensions connaissez-vous ?

Disques et partitions

The screenshot shows the Windows Computer Management console. The left pane displays the navigation tree with 'Disk Management' selected under 'Storage'. The main pane shows a table of volumes and a graphical disk layout below it.

Volume	Layout	Type	File System	Status	Capacity	Free Space	% Free	Fa	Actions
Backup (I:)	Simple	Dynamic	NTFS	Healthy	9,32 GB	8,94 GB	96 %	Nc	Disk Management More Actions
DOCUMENTS (H:)	Simple	Dynamic	NTFS	Healthy	57,27 GB	22,90 GB	39 %	Nc	
Intel 915_925_AI (E:)	Partition	Basic	CDFS	Healthy	624 MB	0 MB	0 %	Nc	
PROGRAMS (D:)	Partition	Basic	NTFS	Healthy	24,41 GB	23,44 GB	96 %	Nc	
SBS2003 (F:)	Partition	Basic	NTFS	Healthy	19,53 GB	10,57 GB	54 %	Nc	
VISTASYSTEM (G:)	Partition	Basic	NTFS	Healthy (Boot, Page File, Crash Dump)	14,65 GB	7,00 GB	47 %	Nc	
XPSYSTEM (C:)	Partition	Basic	NTFS	Healthy (System, Active)	14,65 GB	8,66 GB	59 %	Nc	

Disk	Layout	Type	File System	Status	Capacity	Free Space	% Free	Fa
Disk 0 Basic 232,88 GB Online	XPSYSTEM (C:) 14,65 GB NTFS Healthy (System,)	PROGRAMS (D:) 24,41 GB NTFS Healthy	SBS2003 (F:) 19,53 GB NTFS Healthy	VISTASYSTEM (G:) 14,65 GB NTFS Healthy (Boot, Pa)	159,63 GB	Free space		9 M Un
Disk 1 Dynamic 57,27 GB Online	DOCUMENTS (H:) 57,27 GB NTFS Healthy							
Disk 2 Dynamic 9,32 GB Online	Backup (I:) 9,32 GB NTFS Healthy							
CD-ROM 0 DVD 624 MB Online	Intel 915_925_AI (E:) 624 MB CDFS Healthy							

Legend: ■ Unallocated ■ Primary partition ■ Extended partition ■ Free space ■ Logical drive ■ Simple volume

Disques et partitions

- Chaque disque dur est **partitionné**
- Une partition est un segment du disque, destiné à être **formaté** (= à recevoir un système de fichiers)
- Permet de **cloisonner** les informations, en particulier :
 - lorsqu'on installe plusieurs OS différents
 - lorsqu'on veut séparer données et logiciels (une très bonne idée!)
 - Sous Linux : partitions « swap »
- Chaque partition reçoit un système de fichiers (NTFS, EXT4...) qui lui est propre

Disques et partitions

- Le premier secteur (= le « tout début ») d'un disque dur contient le programme d'amorçage et la table des partitions
- Certaines partitions sont amorçables (*bootables*), d'autres non (= seulement pour les données)
- Autrefois : restriction sur le nombre de partitions primaires, notion de partition secondaire
- Partitionnement du disque : à faire de préférence avant toute installation d'OS. Ensuite, on peut redimensionner certaines partitions pour faire de la place, mais c'est plus délicat.

Disques et partitions

Exemple : disque de 1 To

- 40 Go pour Linux et ses programmes
- 8 Go de « swap »
- 100 Go pour /home (données utilisateur Linux)
- 100 Go pour Windows et ses programmes
- Le reste pour des données Windows (qui seront accessibles depuis Linux)
- Au total, 2 partitions bootables. Un gestionnaire de boot permettra de choisir si on lance Linux ou Windows.

Systemes de fichiers

- On a découpé le disque en partitions = emplacements physiques
- Ces emplacements contiennent des suites de 0 et de 1 codés magnétiquement.
- Comment deviennent-ils des fichiers ?
- Fichiers = suites de blocs (la plus petite unité que le périphérique de stockage est capable de gérer). Le contenu de ces blocs, simple suite de données binaires, peut être interprété selon le format de fichier comme des caractères, des nombres entiers ou flottants, des codes d'opérations machines, des adresses mémoires...
- Subtilité : sous UNIX/Linux, il y a aussi des « fichiers spéciaux » qui désignent des périphériques, des zones de mémoire temporaires, des connexions réseau...

Systemes de fichiers

- C'est ici que le **systeme de fichiers** joue son rôle : il relie un fichier à un emplacement physique du disque (ou à un périphérique, etc)
- Il permet l'accès au contenu des fichiers stockés (l'**ouverture** du fichier, son **enregistrement**, sa **copie** ou son **déplacement** dans un second emplacement, ou sa **suppression**) à partir de leur **chemin d'accès**, formé d'un nom précédé d'une liste de répertoires imbriqués.

Une grande variété !

- Une liste de systèmes de fichiers existants :
https://fr.wikipedia.org/wiki/Syst%C3%A8me_de_fichiers
- Les plus fréquents aujourd'hui :
 - FAT32 (clés USB / cartes SD),
 - NTFS (systèmes Windows),
 - ext4 (Linux),
 - HFS+ (MacOS) qui va être remplacé par APFS.

A titre culturel :

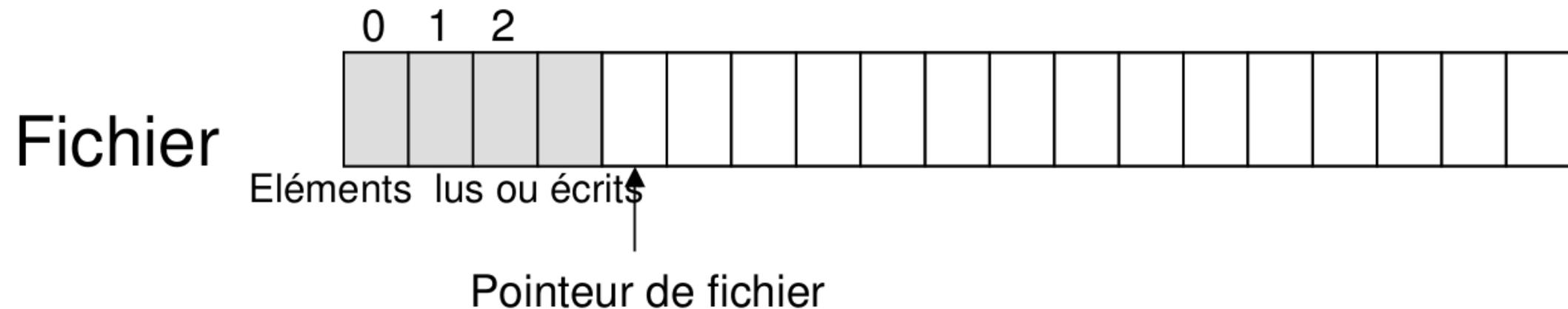
<https://www.macg.co/os-x/2017/01/apfs-le-futur-systeme-de-fichiers-dapple-qui-va-changer-votre-vie-94735>

Systemes de fichiers

- Certains systemes de fichiers gèrent :
 - Des systemes de possession de fichiers et de permissions (qui peut lire / modifier un fichier, qui peut exécuter une application)
 - Des systemes de journalisation, pour pouvoir corriger plus efficacement les erreurs
 - Une réduction de la fragmentation : s'assurer que les données d'un même fichier sont toutes assez proches sur le disque
 - ...

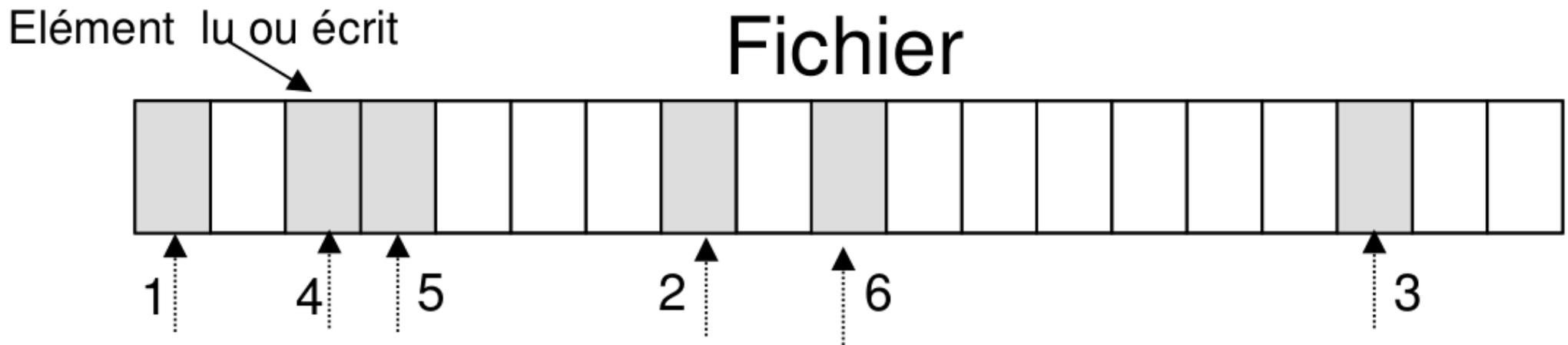
Accès à la mémoire de masse

- Accès **séquentiel** : les données sont « dans l'ordre »
- Adapté aux mémoires séquentielles telles que les bandes magnétiques



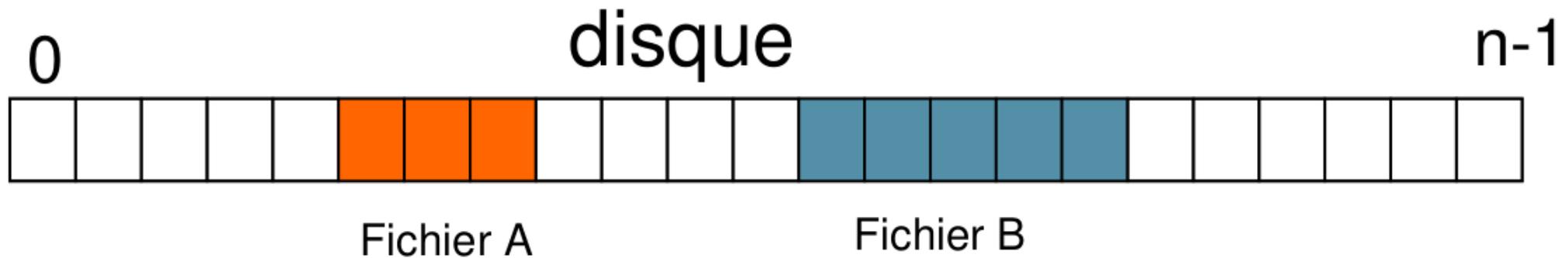
Accès à la mémoire de masse

- **Accès direct (ou accès aléatoire)** : on choisit à quelle case mémoire accéder. Cf disques durs.



Méthodes d'allocation de fichiers

- **Allocation contigüe** : chaque fichier occupe des blocs contigus sur le disque

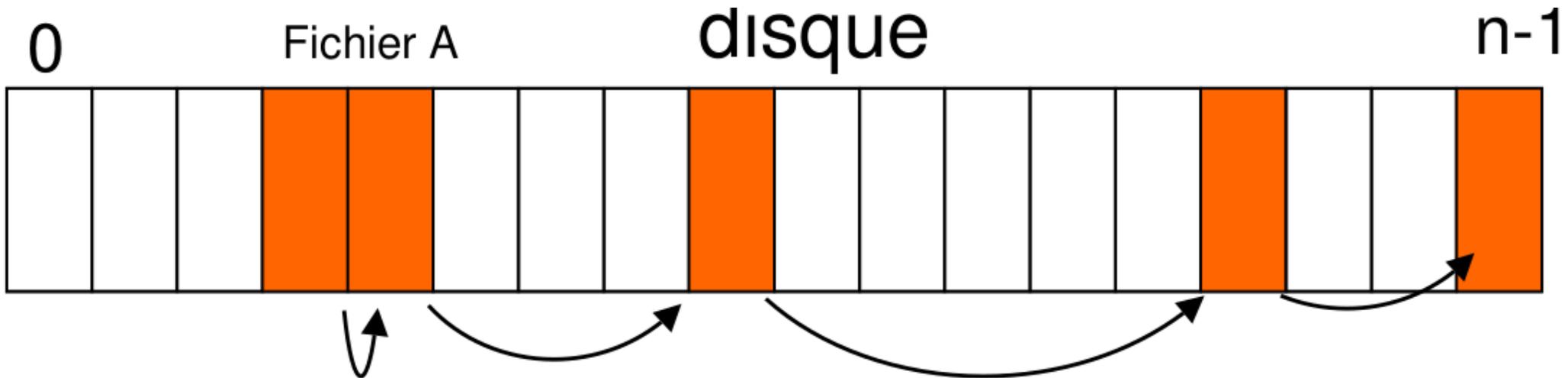


Méthodes d'allocation de fichiers

- Avantages :
 - Facile à implémenter
 - Accès direct aux blocs en temps constant
 - Adapté aux supports « Write Once » (CD/DVD /BD)
- Inconvénients :
 - Extension d'un fichier : pas sûr d'avoir assez de place, il peut falloir le déplacer...
 - Fragmentation : « trous » entre les fichiers ; perte de place

Méthodes d'allocation de fichiers

- **Allocation chaînée** : un fichier occupe une **liste chaînée** de blocs répartis sur le disque
- Chaque bloc contient une donnée + l'adresse du prochain bloc



Méthodes d'allocation de fichiers

- Avantages :
 - Possibilité d'étendre facilement un fichier
 - Allocation par bloc individuel : pas affecté par la fragmentation (la performance l'est cependant)
- Inconvénient :
 - Inadapté à l'accès direct : pour accéder à un bloc, on doit lire tous ceux qui précèdent
 - Les pointeurs sont stockés au beau milieu du disque, mélangés avec les données

La FAT

- Séparer les pointeurs et les données : idée de la **FAT** (File Allocation Table)
- A chaque bloc est associé une entrée dans la FAT qui indique le numéro du bloc suivant

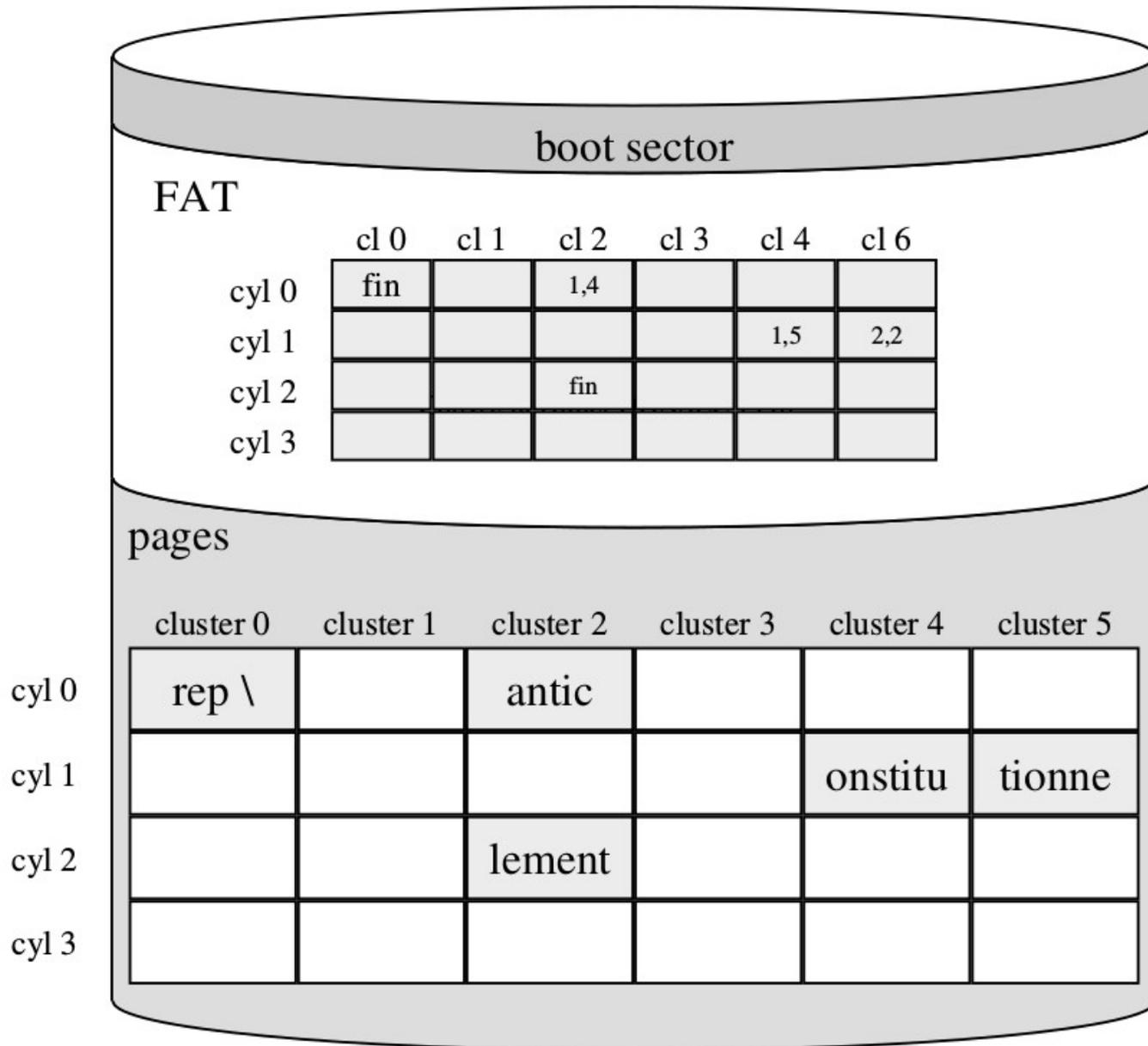
0	3
1	Fin de fichier
2	4
3	6
4	1
5	
6	Fin de fichier
	▪
	▪
	-

Fichiers	blocs occupés
A	0 3 6
B	2 4 1

La FAT

- Avantages :
 - Extension facile des fichiers
 - Pointeurs au début du disque, dans la FAT, pas au milieu des données
 - Accès direct facile
- Inconvénients :
 - Occupation de mémoire centrale par la FAT, taille de la FAT (pour un disque de 1 Go en blocs de 1 Ko : la FAT prend 4 Mo)
 - Pas compatible avec les systèmes de fichiers de plus de 2 Go. Extensions : FAT16, FAT32 (dans FAT32, systèmes plus gros supportés, mais fichiers de taille 4 Go max)

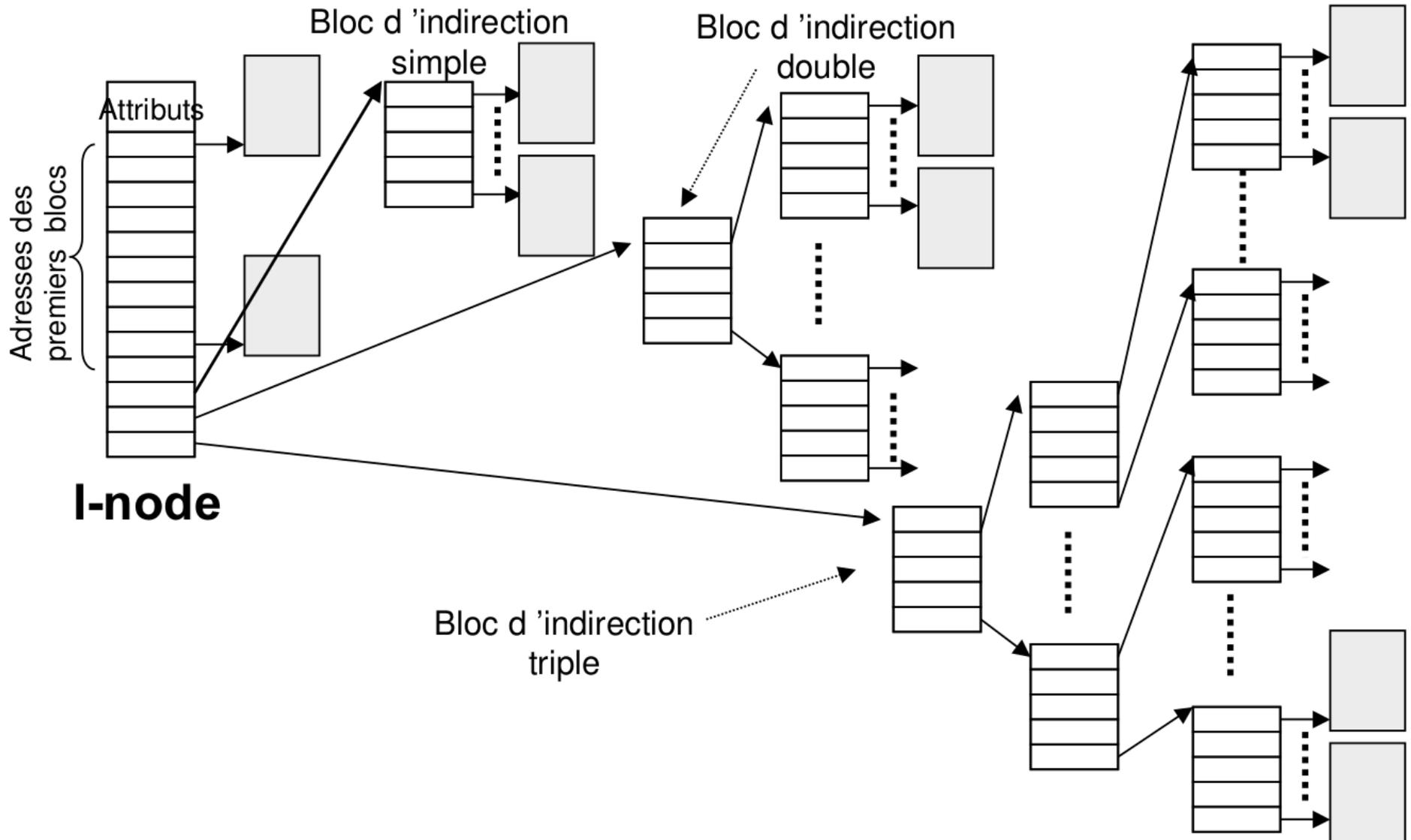
La FAT



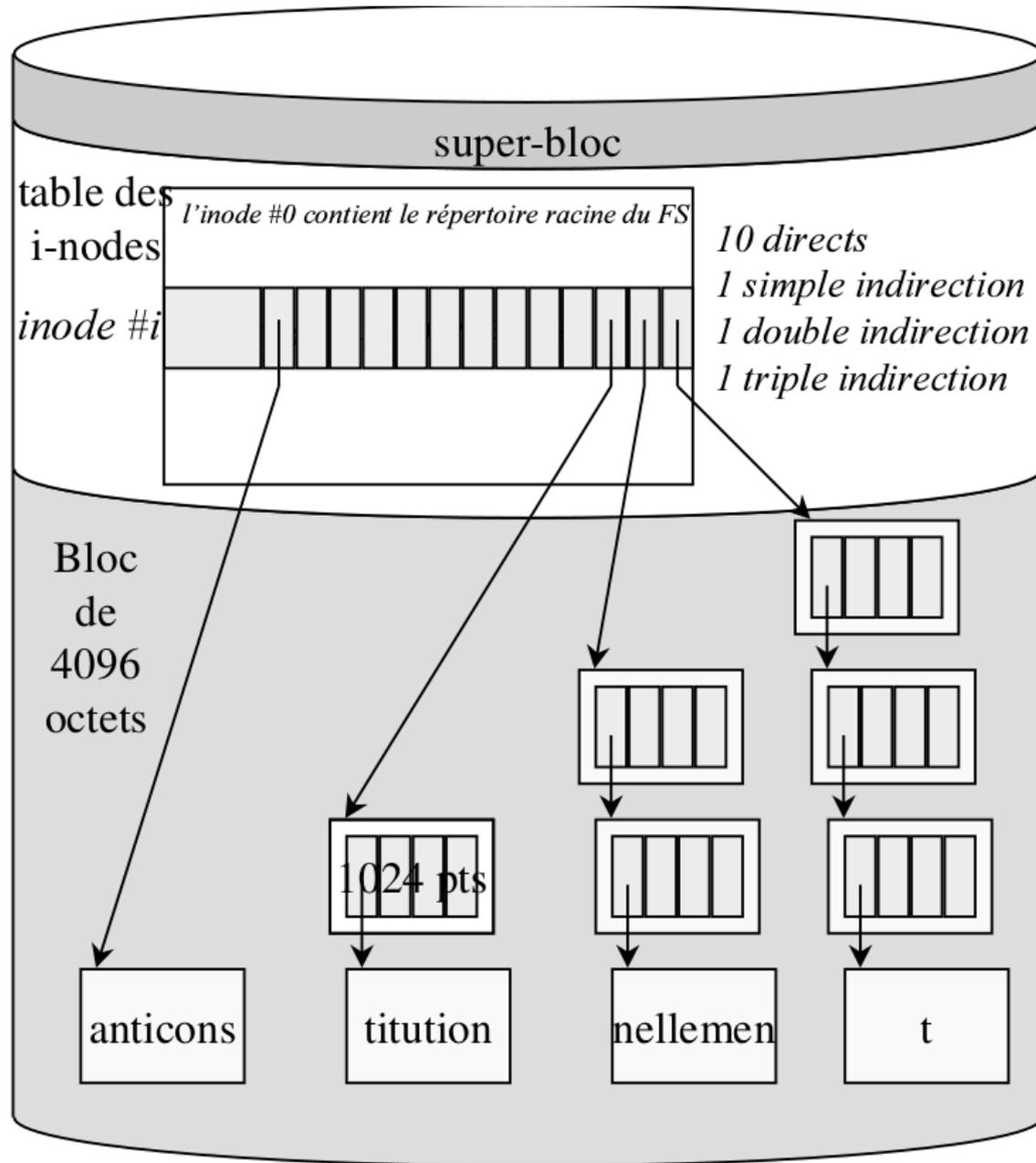
inodes

- Idée : éclater la FAT en petites tables d'information appelées inodes
- Solution notamment sous Unix (ext2 et suivants)
- A chaque fichier est associé un inode, contenant des méta-données : type, propriétaire, groupe, droits d'accès, nombre de pointeurs vers cet inode, les blocs où les données du fichiers sont stockées, . . . , mais pas le nom du fichier
- La table des inodes est hiérarchisée

inodes



Inodes dans ext2



Inodes

- Nombre d'inodes fixé à la création du système de fichiers. Détermine donc un nombre max de fichiers.
- Cette structuration permet d'allouer jusqu'à environ 16 millions de blocs à un fichier. Si on choisit des blocs de 4Ko, fichiers de 64 Go max (là encore on peut choisir la taille des blocs mais uniquement au moment du formatage)

Inodes

- Contrairement à la FAT qui doit être chargé entièrement dans la mémoire centrale, il suffit de charger les inodes des fichiers ouverts
- Allocation facile : par blocs individuels
- Accès direct facile (au maximum 4 accès disque)
- Adapté aux disques de grande capacité
- Autre solution pour les gros disques : NTFS (Windows). Utilise les arbres B+ (plus compliqués, on n'en parlera pas ici)

Inodes

- Sous Linux, « ls -i » donne l'information sur l'inode. « stat nom_fichier » donne des informations sur l'accès, etc
- **Dossiers** : ce sont des fichiers spéciaux. Ils contiennent une liste de noms et d'inodes (désignant les fichiers contenus dans le dossier)

Propriétaire et droits d'accès

- Avec « ls -l » on peut connaître :
 - L'utilisateur propriétaire d'un fichier
 - Le groupe propriétaire d'un fichier
 - Les permissions : 10 caractères
 - Le premier : « d » si répertoire (directory), « - » sinon
 - De 2 à 4 : permissions de l'utilisateur propriétaire :
 - « r » signifie « read »
 - « w » signifie « write »
 - « x » signifie « execute »
 - De 5 à 7 : permissions des utilisateurs faisant partie du groupe propriétaire
 - De 8 à 10 : permissions des autres utilisateurs

Modifier les propriétaires / permissions

- Modifier le propriétaire : avec « chown »
- Modifier le groupe : avec « chgrp »
- Modifier les permissions : avec « chmod ».
Plusieurs syntaxes, notamment : « chmod XYZ nom_fichier » où X, Y et Z sont trois nombres entre 0 et 7

Modifier les propriétaires / permissions

- Exécuter vaut 1
- Modifier vaut 2
- Lire vaut 4
- On fait la somme des permissions. Par exemple : lire et exécuter mais pas modifier vaut 5.
- Ensuite, XYZ est composé de X = la somme pour l'utilisateur, Y = la somme pour le groupe, Z = la somme pour les autres
- Exemple : « `chmod 755 fichier_executable` »

Modifier les propriétaires / permissions

- Possible aussi sous Windows, via l'interface graphique

