

5. Organisation des mémoires de masse

- **2 types de mémoires de masse :**
 - **Mémoires de travail : rapides (ms), de taille moyenne (disques durs, mémoires flash...)**
 - **Mémoires de sauvegarde : capacités importantes, mais lentes (CD, DVD, bande magnétique...)**
- Cartouche à bande magnétique **LTO 5 (Linear Tape Open)** débit 140 Mo/s (280 Mo/s en compression), capacité 3 To
- **LTO 7 (2015) : 300 Mo/s (750 Mo/s en compression), capacité 15 To**



5. Organisation des mémoires de masse

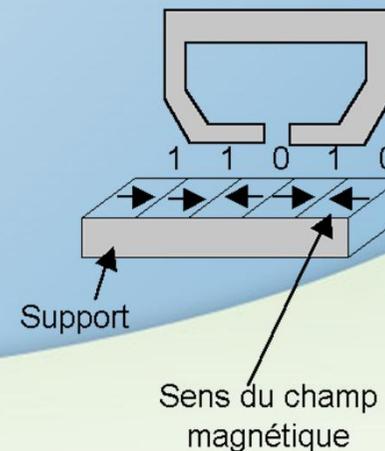


Le disque dur

- **Disque dur** = support d'enregistrement magnétique
- **Composé de** plusieurs disques rigides (« durs ») en métal, verre ou céramique = **plateaux**, empilés les uns sur les autres et qui tournent autour d'un axe (vitesse de rotation : 4000 à 15000 tr/min, voire +)
- **Les informations** sont **stockées** sous forme **analogique** sur des fines couches magnétiques déposées sur les deux surfaces des plateaux = **pistes**
- La **lecture/écriture** se fait par **des têtes** placés des **deux côtés de chaque plateau**



Tête de lecture / écriture



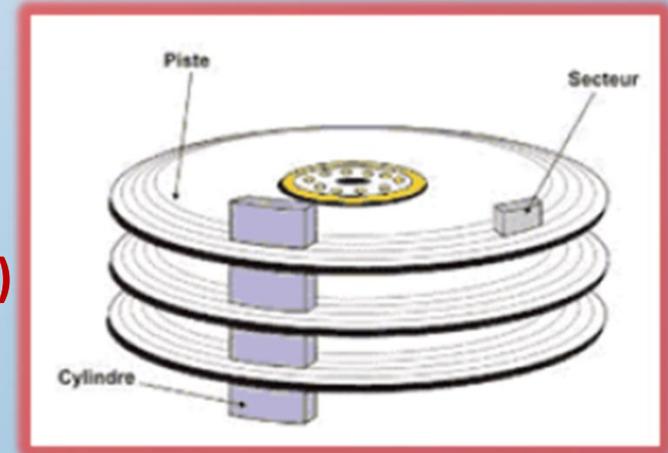
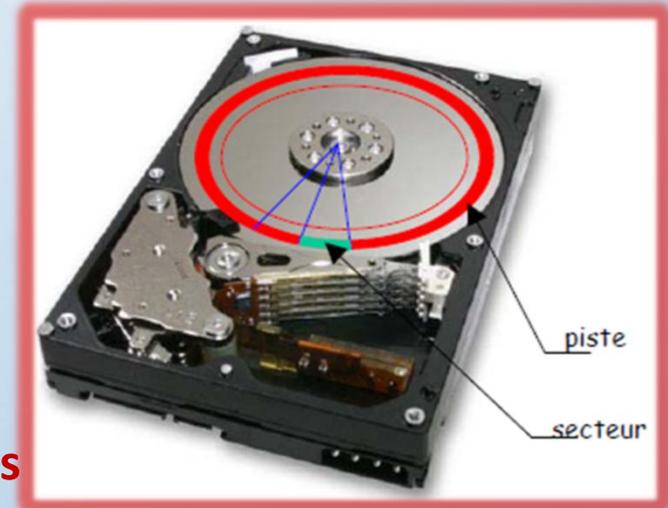
5. Organisation des mémoires de masse



Le disque dur

- **Structure de l'information**

- Sur chaque plateau, les données sont disposées sur des **pistes concentriques** (« tracks ») = couronnes circulaires
- L'ensemble des **pistes de même diamètre de tous les plateaux = cylindre** (« cylinder »)
- Chaque piste (cylindre) est numérotée, en partant de l'extérieur du plateau (début : « track 0 » ou « **cylinder 0** » - **près du bord**)
- Chaque piste (cylindre) est divisé en **secteurs (blocs)** (« sector », « block »)
- Chaque **secteur est numéroté**, en commençant par 1



No. de cylindre (piste) + No. et face de plateau + No. secteur
= possibilité de localiser clairement une zone du disque

- Exemple d'adressage : Cylindre (piste) 0, Plateau 2 face interne, secteur 12



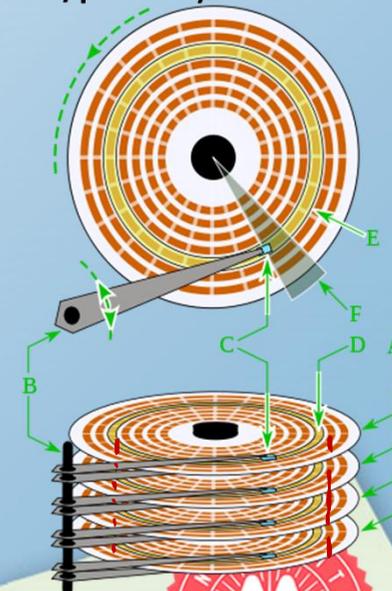
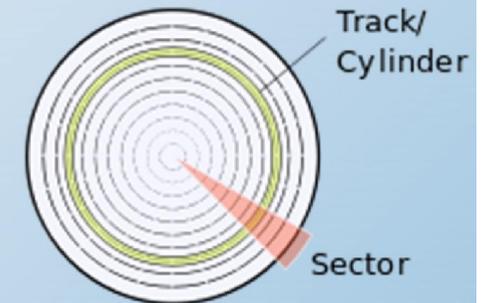
Le disque dur

- Structure de l'information
- Dans la pratique, on **numérote aussi les têtes**, en **commençant par 0**
- On peut donc accéder à un secteur du disque dur à l'aide :
 - du numéro de **cylindre** $C \in [0, NC-1]$ (NC = nb total de cylindres)
 - du numéro de **tête** H (« head ») $\in [0, NT-1]$ (NT = nb. total de têtes)
 - du numéro de **secteur** $S \in [1, NS]$ (NS = nb total de secteurs/plateau)



Adressage **CHS** (Cylinder/Head/Sector)

- Exemples :
 - Secteur 0/0/1 = 1^{er} secteur accédé par la première tête positionnée sur le premier cylindre... ensuite la tête accède au secteur 0/0/2... jusqu' à 0/0/NS à la fin d'une rotation complète
 - Secteur suivant : 0/1/1 – accédé par la tête 1... jusqu'à 0/1/NS ... et ainsi de suite...
 - Adresse du tout dernier secteur du disque : NC-1/NT-1/NS



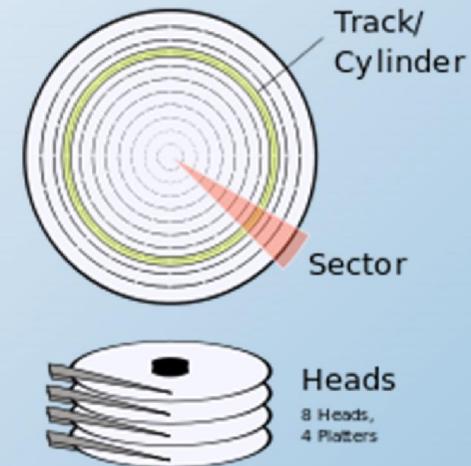
5. Organisation des mémoires de masse



Le disque dur

- **Limites de l'adressage CHS**

- Le **BIOS** code :
 - le numéro de **cylindre C** sur 10 bits
 - le numéro de **tête H** sur 8 bits
 - le numéro de **secteur S** sur 6 bits



Max adressable : $2^{10} * 2^8 * 2^6 = 1024 * 256 * 64$ secteurs

En général, un secteur = 512 octets

Donc **taille max d'un disque adressé par le BIOS en adressage CHS :**

$1024 * 256 * 64 * 512 \sim 8 \text{ Go}$



- **Nécessité de trouver d'autres types d'adressage pour gérer des disques + gros (ECHS, MFM, LBA)**
- **Nécessité de faire évoluer le BIOS → UEFI**

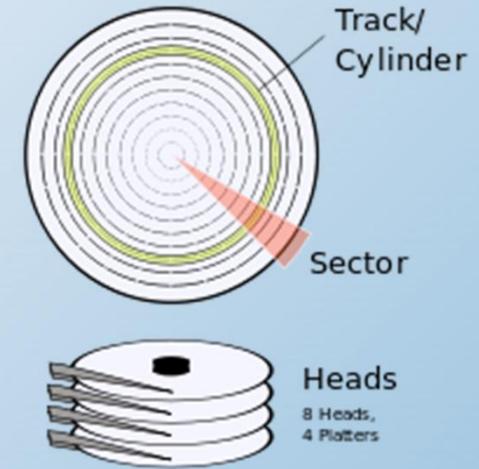


Le disque dur

- Limites de l'adressage CHS

- **Néanmoins :**

- **L'adressage CHS reste encore employé dans les premières phases de démarrage d'un ordinateur, puisqu'il permet toujours d'accéder aux premiers secteurs d'un disque !**
- **Ainsi, le BIOS charge le secteur 0 / 0 / 1 du premier disque dur, qui est souvent un MBR ; ce dernier emploie à son tour une adresse CHS pour charger le secteur d'amorçage de la partition active**
- C'est l'emploi d'adresses CHS durant cette phase qui fait que certains utilitaires disques vous alertent si la partition active se trouve au-delà des premiers 8 Go d'un disque...



car le secteur d'amorce deviendrait alors inaccessible par une adresse CHS !



Le disque dur

- **Adressage LBA (« Logical Block Addressing »)**

- Permet de désigner d'une façon unique un secteur d'un disque
- Idée : masquer totalement la géométrie du disque aux programmes qui l'utilisent
- Chaque secteur est donc désigné par un numéro unique $A = 0.. N-1$ avec $N = \text{nb. total des secteurs du support}$
- Puisque N ne dépend plus de la géométrie du support
 - possibilité d'utiliser le LBA quelque soit le type de support, pas seulement des disques durs
 - emploi avec des clés USB, bandes magnétiques etc !
- Possibilité également de gérer un nombre plus important de secteurs
 - des disques durs plus grands
- Pour les disques durs :
 - conversion CHS/LBA : $A = (C \times NT \times NS) + (H \times NS) + S - 1$
 - conversion LBA/CHS :
 - $S = (A \% NS) + 1$
 - $H = (A - S + 1) / NS \% NT$
 - $C = (A - S + 1) / NS \div NT$

5. Organisation des mémoires de masse



Le disque dur

- **Unité d'allocation**

- **Unité d'allocation (« cluster »)**

= zone minimale que peut occuper un fichier sur le disque (1 à 16 secteurs) = « cellule de mémoire » du disque

- Taille minimale d'un cluster = **1 secteur = le plus souvent 512 octets**

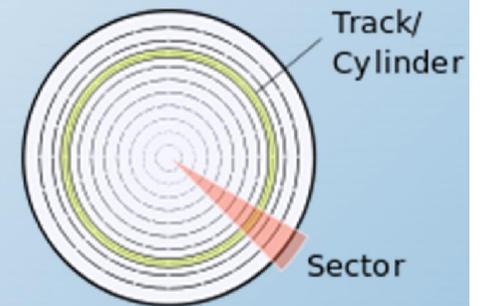
- Intérêt de la « **clusterisation** » :

compromis temps d'accès/taux de remplissage du disque

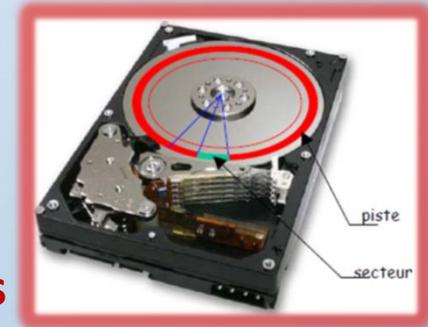
- clusters **petits** → utilisation parcimonieuse du disque dur, mais accès relatif + lent → intéressant pour les petits disques

- clusters **grands** → diminution du temps d'accès au fichier (car moins de clusters sur le disque), mais « gaspillage » d'espace disque (si la taille d'un fichier n'est pas un multiple de la taille d'un cluster, le cluster contenant la fin du fichier ne sera occupé que partiellement)

- **Tailles usuelles des clusters : 512 à 2048 octets, en fonction de la taille du disque dur ou du média**

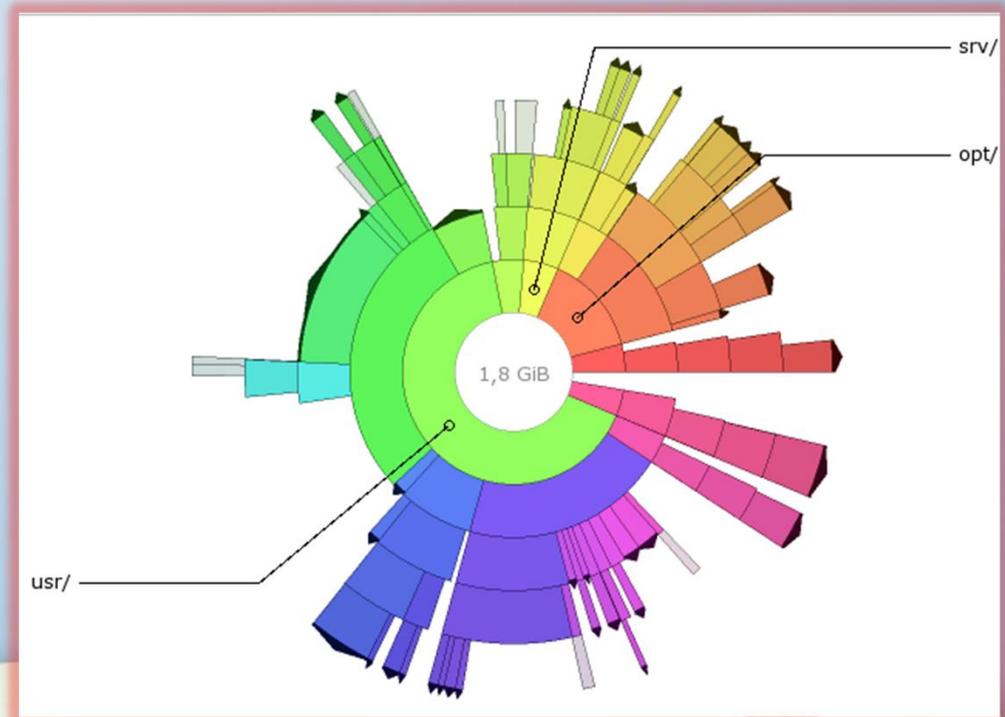


Le disque dur



- **Formatage et partitionnement**

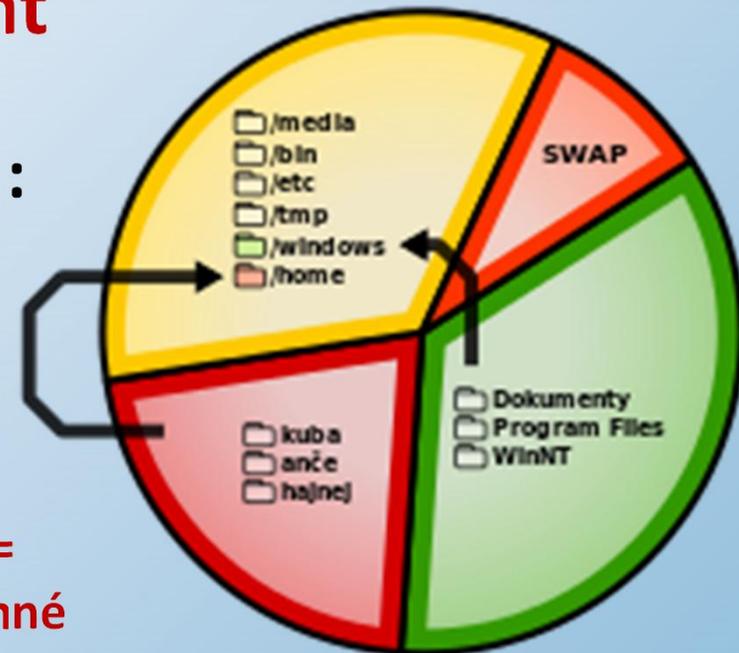
- **Le nombre de pistes / plateau (nb. de cylindres) et de secteurs** est déterminé en usine et les pistes sont réalisés physiquement, par dépôt des couches magnétiques sur les plateaux = **formatage physique (bas niveau)**
- Le formatage physique **ne peut être modifié**
- **Formatage logique (haut niveau) :** écriture du secteur de démarrage et création du système de fichiers
- Le formatage logique **dépend du système d'exploitation utilisé**



5. Organisation des mémoires de masse

Le disque dur

- **Formatage et partitionnement**
- **Formatage logique (haut niveau) :**
- **Plusieurs systèmes de fichiers peuvent cohabiter sur un même disque dur**
- **Tout disque dur peut être divisé en partitions = fractions contenant un système de fichiers donné**
- **Partitionnement = fractionnement d'un disque dur réel (matériel) en plusieurs disques virtuels (logiciels)**
- **Un disque dur peut contenir une ou plusieurs partitions**
- **S'il en contient plusieurs, l'OS les voit comme des disques différents**

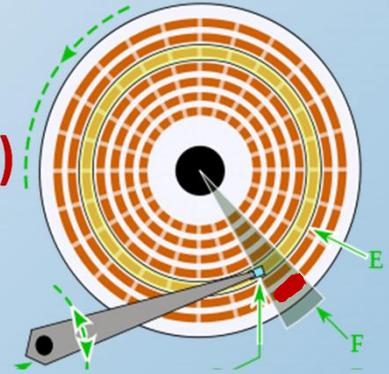


Le disque dur

- **Le Master Boot Record (MBR ou « zone amorce »)**

- **MBR = le 1^{er} secteur adressable d'un disque dur**

(CHS 0/0/1, ou LBA 0)



- **Taille : 512 octets**

- **Il contient :**

- **une routine d'amorçage** dont le but est de **charger le système d'exploitation (« OS »)** ou le « boot loader » (chargeur d'amorçage, s'il existe) **présent sur la partition active**

- **la TABLE des PARTITIONS = description des diverses partitions présentes sur le disque, la partition active étant indiquée**

- **En raison de sa taille limitée à 512 octets, la routine d'amorçage nécessitant de 440 à 446 octets et la fin du MBR étant indiqué par 2 octets,**

il ne reste que 64 octets disponibles pour la table des partitions dans le MBR

- **La description de chaque partition nécessitant 16 octets,**

le MBR peut donc gérer maximum 4 partitions = partitions primaires



Le disque dur

- **Le Master Boot Record (MBR ou « zone amorce »)**

Structure du Master Boot Record :

Adresse		Description	Taille en octets
Hex	Déc		
0000	0	Routine	440 (max. 446)
01B8	440	Signature optionnelle	4
01BC	444	Habituellement nul ; 0x0000	2
01BE	446	Table des partitions primaires (Quatre entrées de 16 octets (IBM Partition Table scheme))	64
01FE	510	55h	2
01FF	511	AAh	
Taille totale du MBR : 440 + 4 + 2 + 64 + 2 =			512

- **A l'octet 510, le mot hexa AA55 = « magic number » doit impérativement être présent pour que l'amorçage ait lieu!**

Séquence d'amorçage (« boot ») :

- Après la phase de test du BIOS, le BIOS lit le 1^{er} secteur des périphériques amorçables qui ont été définis par l'utilisateur à l'aide du SETUP du BIOS (section BOOT DEVICE ORDERING)
- **Lorsqu'il trouve un périphérique contenant le « magic number » 0xAA55, il charge le code d'amorçage à l'adresse mémoire 0x7C00 et l'exécute**
- La main est alors donnée au chargeur d'amorçage (« boot strap loader ») par ce code d'amorçage

Le disque dur

- Le Master Boot Record (MBR ou « zone amorce »)

Identificateurs de partition (File System Descriptor)

- Sur un ordinateur de type PC, un identificateur de 1 octet associé à chaque partition permet de connaître a priori quel type de système de fichier elle abrite
- Exemples** d'identificateurs de fichiers :

ID	Type
00	Vide
01	FAT12
02	XENIX root
03	XENIX /usr
04	FAT16 <32Mio (adressage CHS)
05	Étendue
06	FAT16
07	NTFS (et son prédécesseur HPFS)
08	AIX , voir JFS
09	AIX bootable
0a	OS/2 Boot Manager
0b	Win95 OSR2 FAT32 (adressage CHS)
0c	Win95 OSR2 FAT32 (adressage LBA , appelée aussi FAT32X)
0e	Win95 FAT16 (adressage LBA)
0f	Étendue (adressage LBA)

80	Ancien Minix
81	Minix / ancien Linux
82	Swap Linux / pool ZFS
83	Ce type de partition est utilisé par les systèmes de fichiers ext2 , ext3 , ext4 , ReiserFS et JFS (Linux en particulier)
84	OS/2 hidden C:
85	Linux étendu
86	NTFS volume set
87	NTFS volume set
8e	Linux LVM
93	Amoeba
94	Amoeba BBT
9d	SDF
9f	BSD/OS

Le disque dur

- Le Master Boot Record (MBR ou « zone amorce »)

- Détails et exemples des 16 octets = caractéristiques d'une partition dans le MBR

Offset	valeur dans mon tableau	description	signification
1BE	80	partition bootable ? (80 = oui, 00 = non)	oui
de 1BF à 1C1	01 01 00	Secteur de début de la partition (en CHS), le 1er octet codant pour la tête (H) : 01 = 1 les 2 autres sont regroupés pour donner la valeur secteur (S) et cylindre (C)	C H S 0 1 1
1C2	7	Type de la partition (NTFS, FAT, Linux,...)	NTFS
de 1C3 à 1C5	FE FF FF	Secteur de fin de la partition (en CHS), le 1er code pour la tête (H) : FE = 254, les 2 autres sont regroupés pour donner la valeur secteur (S) et cylindre (C)	C H S 1023 254 63
de 1C6 à 1C9	3F 00 00 00	Nombre de secteurs avant la partition. Les 4 octets sont à lire à l'envers : 00 00 00 3F = 63 secteurs "before"	63
de 1CA à 1CD	5E 38 27 03	Nombre total de secteurs dans la partition. Les 4 octets sont à lire à l'envers : 03 27 38 5E = 52 901 982 secteurs	52 901 982

5. Organisation des mémoires de masse



Le disque dur

- Pour chaque partition, un octet indique si elle est « bootable » ou pas : « **boot flag** »
 - « Boot flag » = **80** → **partition amorçable**
 - « Boot flag » = **00** → **partition pas amorçable**

Exemple de MBR

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F		
0000000000	B3	C0	8E	D0	BC	00	7C	8E	C0	8E	D8	BE	00	7C	BF	00	SAZDw. ŽAZO%. ž.	
0000000010	06	B9	00	02	FC	F3	A4	50	68	1C	06	CB	FB	B9	04	00	. ' . . uóPh. . EÜ² . .	
0000000020	BD	BE	07	80	7E	00	00	7C	0B	0F	85	10	01	83	C5	10	¼%. €~. fÄ.	
0000000030	E2	F1	CD	18	88	56	00	55	C6	46	11	05	C6	46	10	00	añí. ^V.UEF. . EF. .	
0000000040	B4	41	BB	AA	55	CD	13	5D	72	0F	81	FB	55	AA	75	09	'A»^Uí. r. . úU^u.	
0000000050	F7	C1	01	00	74	03	FE	46	10	66	60	80	7E	10	00	74	÷Á. . t. pF. f' e~. . t	
0000000060	26	66	68	00	00	00	00	66	FF	76	08	68	00	00	68	00	&fh. . . . fyv. h. . h.	
0000000070	7C	68	01	00	68	10	00	B4	42	8A	56	00	8B	F4	CD	13	h. . h. . BŠV. < óí.	
0000000080	9F	83	C4	10	9E	EB	14	B8	01	02	BB	00	7C	8A	56	00	ÝfÄ. žē. ŠV.	
0000000090	8A	76	01	8A	4E	02	8A	6E	03	CD	13	66	61	73	1E	FE	Šv. ŠN. Šn. í. fas. p	
00000000A0	4E	11	0F	85	0C	00	80	7E	00	80	0F	84	8A	00	B2	80	N. €~. €. . . Š. ^e	
00000000B0	EB	82	55	32	E4	8A	56	00	CD	13	5D	EB	9C	81	3E	FE	ē, U2āŠV. í. ēē. > p	
00000000C0	7D	55	AA	75	6E	FF	76	00	E8	8A	00	0F	85	15	00	B0	} U^unýv. ēŠ. °	
00000000D0	D1	E6	64	E8	7F	00	B0	DF	E6	60	E8	78	00	B0	FF	E6	Ňadē. . . ^æ^ēx. °yē	
00000000E0	64	E8	71	00	B8	00	BB	CD	1A	66	23	C0	75	3B	66	81	dēq. í. f#Äu; f.	
00000000F0	FB	54	43	50	41	75	32	81	F9	02	01	72	2C	66	68	07	úTCPAu2. ú. . r, fh.	
0000000100	BB	00	00	66	68	00	02	00	00	66	68	08	00	00	00	66	».. fh. . . . fh. . . . f	
0000000110	53	66	53	66	55	66	68	00	00	66	68	00	66	68	00	7C	00	SfSfUfh. . . . fh. .
0000000120	00	66	61	68	00	00	07	CD	1A	5A	32	F6	EA	00	7C	00	. fah. . . . í. Z2ōē. .	
0000000130	00	CD	18	A0	B7	07	EB	08	A0	B6	07	EB	03	A0	B5	07	. í. . . ē. q. ē. v.	
0000000140	32	E4	05	00	07	8B	F0	AC	3C	00	74	FC	BB	07	00	B4	2ä. . . < ē- < . tú. . .	
0000000150	0E	CD	10	EB	F2	2B	C9	E4	64	EB	00	24	02	E0	F8	24	. í. ēō+ēādē. ē. āō\$	
0000000160	02	C3	49	6E	76	61	6C	69	64	20	70	61	72	74	69	74	. ĀInvalid partit	
0000000170	69	6F	6E	20	74	61	62	6C	65	00	45	72	72	6F	72	20	ion table. Error	
0000000180	6C	6F	61	64	69	6E	67	20	6F	70	65	72	61	74	69	6E	loading operatin	
0000000190	67	20	73	79	73	74	65	6D	00	4D	69	73	73	69	6E	67	g system. Missing	
00000001A0	20	6F	70	65	72	61	74	69	6E	67	20	73	79	73	74	65	operating syste	
00000001B0	6D	00	00	00	00	62	7A	99	49	95	43	0A	00	00	00	01	m. bz^I^C.	
00000001C0	01	00	0F	03	3F	04	3F	00	00	00	86	39	01	00	00	00	. . P. ? . ? . . . t9. . ē.	
00000001D0	15	00	07	F6	12	64	00	40	01	00	00	80	17	00	00	F6 d. d. @. . . . ē. . . .	
00000001E0	13	64	07	FE	FF	FF	00	C0	18	00	00	20	84	16	00	FE	. d. pýý. Ä. p	
00000001F0	FF	FF	0F	FE	FF	FF	00	E0	9C	16	00	00	A6	0E	55	AA	ýý. pýý. āē. U^	
0000000200	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000210	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000220	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

File System Descriptors

Indiquer la partition bootable, son système de fichiers et sa taille en Mo.
 c'est la 2, type NTFS, taille = 00 8Q 17 DA

partition 1
 partition 2

DE = Dell

boot flag = 0 (pas bootable)

boot flag

nb. secteurs

07 = NTFS

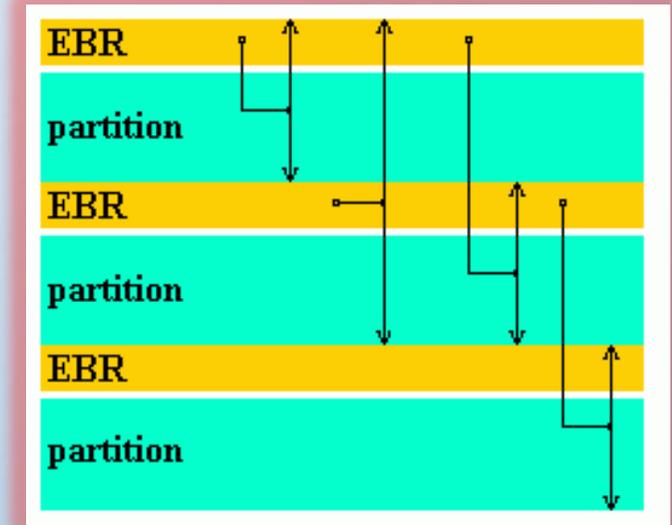
Le disque dur

- **Partitions étendues et logiques (secondaires)**
- Moyen d'étendre la limitation du MBR à 4 partitions
- L'une des 4 partitions peut être désignée comme **partition étendue = partition primaire spéciale** qui va contenir **des partitions secondaires, ou logiques**
- La partition étendue peut contenir jusqu'à 24 partitions logiques, dont les détails sont listés dans la **propre table de partitions de la partition étendue = EBR (Extended Boot Record) ou EPBR (Extended Partition Boot Record)**

Le disque dur

- **Partitions primaires et étendues**

- L'EBR se situe donc en tête d'une partition étendue, sur le disque dur
- Son contenu est identique à celui du MBR
- **L'EBR peut contenir à son tour une partition étendue**, qui contiendra à ce moment-là une **partition tertiaire et ainsi de suite...**
- On dit ainsi que **les EBR sont chaînées** : si la deuxième entrée est non nulle, elle pointe sur le prochain EBR et ainsi de suite
- **Les systèmes d'exploitation (Windows, Linux...)** ne font pas de distinction entre partition primaire et étendue
- En revanche, **le BIOS ne reconnaît directement que les partitions primaires**
- Dans le MBR, **une partition étendue a pour descripteur 05 ou 0F**
- **L'utilitaire fdisk** peut gérer jusqu'à 3 partitions principales + 56 étendues



Le disque dur

- Exemple de EBR

Offset	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
00000000	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000010	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000070	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000090	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000100	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000110	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000120	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000130	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000140	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000150	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000160	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000170	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
00000190	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000001A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000001B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	01	
000001C0	C1	FF	07	FE	FF	FF	3F	00	00	00	4E	5C	0F	07	00	00	Á ŷ . þ ŷ ŷ ? . . . N \
000001D0	C1	FF	05	FE	FF	FF	8D	5C	0F	07	09	28	CE	00	00	00	Á ŷ . þ ŷ ŷ □ \ . . . (î
000001E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000001F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	55	AA	U ²

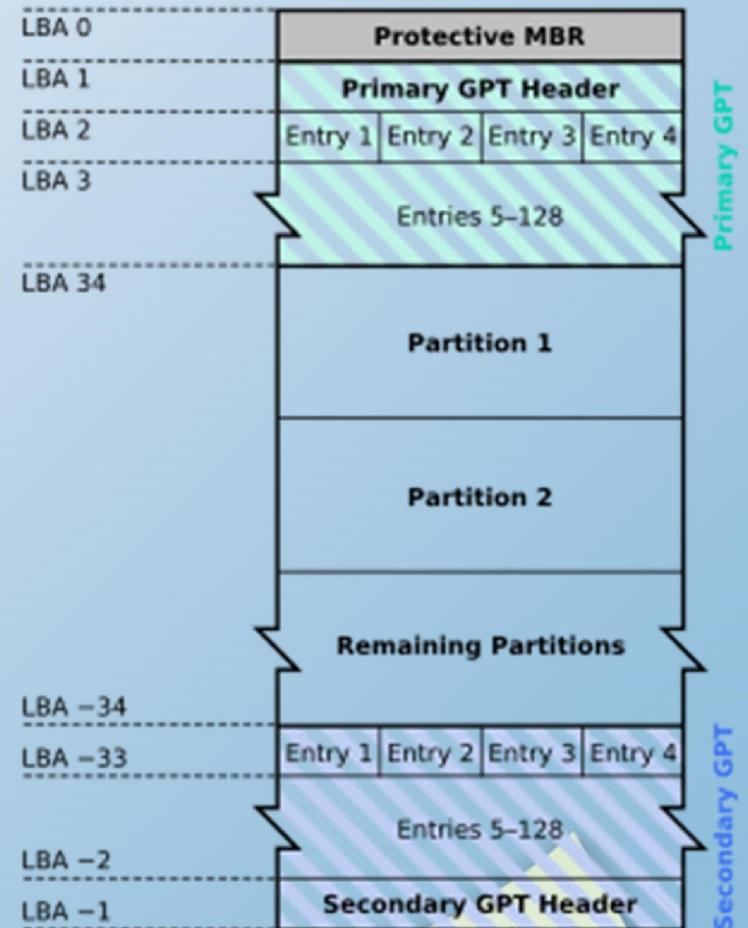
Limitations du MBR

- Nombre de secteurs de la partition codé sur 4 octets (**32 bits**)
→ **2^{32} secteurs adressables**
 - Comme le début de la partition (nombre de secteurs avant la partition) est indiqué de la même façon, il est impossible de démarrer une seconde (ou troisième, etc.) partition derrière une première partition : on sort de la limite
- Limite réelle du MBR, avec des **secteurs de 512 octets** :
- $$2^{32} \text{ secteurs} \times 512 \text{ o/secteur} = 2\,199\,023\,255\,552 \text{ octets} = \mathbf{2,2 \text{ To}}$$

Le disque dur

- **Successeur du MBR : GPT** (lié au UEFI)
Globally Unique Identifier Partition Table
- Utilise l'adressage LBA
- On indique le LBA pour le début et la fin de chaque partition, codée sur 8 octets = **64 bits** (au lieu de 4 pour le MBR)
→ 2^{64} secteurs adressables → avec 512 o/secteur la taille d'une **partition peut faire 8 Zo**
- On peut décrire jusqu'à 128 partitions
- Une copie du GPT est écrite en fin de disque
- Un **MBR protecteur décrivant un disque (de type 0xEE)** est toujours présent sur le secteur 0, indiquant une partition qui recouvre tout le disque
- Les systèmes ou logiciels qui ne connaissent pas GPT voient le disque complètement plein ayant une seule partition d'un type inconnu et refusent de modifier le disque à moins d'effacer cette partition → protection

GUID Partition Table Scheme



5. Organisation des mémoires de masse

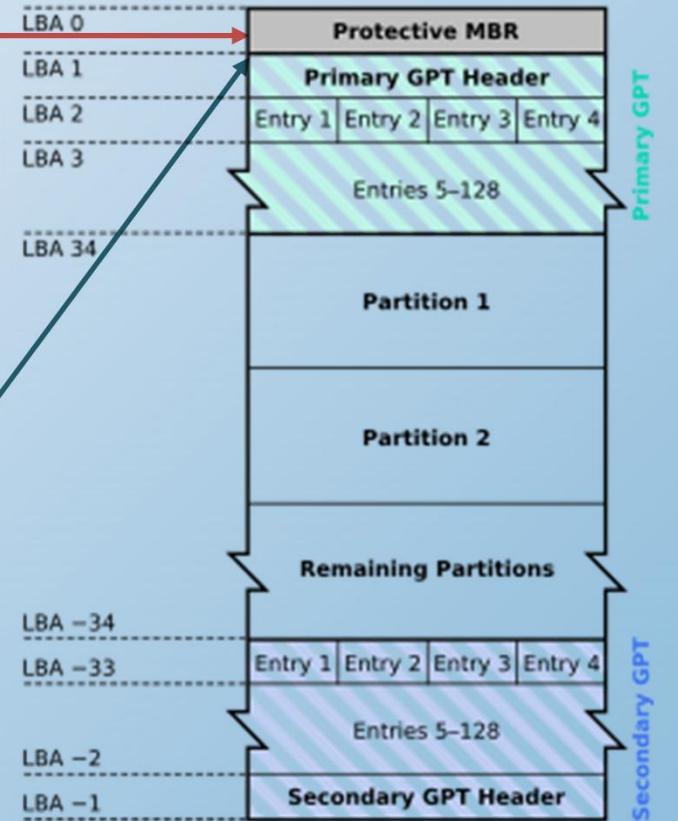


Le disque dur

- Exemple de GPT

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
000000000	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00 Secteur 0
000000010	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000020	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000030	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000040	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000050	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000060	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000070	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000090	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000000F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000100	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000110	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000120	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000130	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000140	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000150	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000160	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000170	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000180	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000190	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000001A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000001B0	00	00	00	00	00	00	00	85	98	9C	0B	00	00	00	00	00
0000001C0	02	00	EE	FF	FF	FF	01	00	00	00	FF	FF	FF	FF	00	00	...iyyy...iyyy..
0000001D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000001E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000001F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	55	AAU*
000000200	45	46	49	20	50	41	52	54	00	00	01	00	5C	00	00	00	EFI PART. \...
000000210	FF	5F	FC	CD	00	00	00	00	01	00	00	00	00	00	00	00	ÿüf.....
000000220	AF	12	9E	3B	00	00	00	00	22	00	00	00	00	00	00	00	—ž;...."
000000230	8E	12	9E	3B	00	00	00	00	7D	A7	B4	9D	8E	41	47	49	ž;....}S'.ŽAGI
000000240	91	47	BC	3C	14	EC	3B	32	02	00	00	00	00	00	00	00	'G<.i;2.....
000000250	80	00	00	00	80	00	00	00	CB	78	D9	E3	00	00	00	00	€...€...ÈxÜä....
000000260	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000270	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000280	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
000000290	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000002A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	
0000002B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	

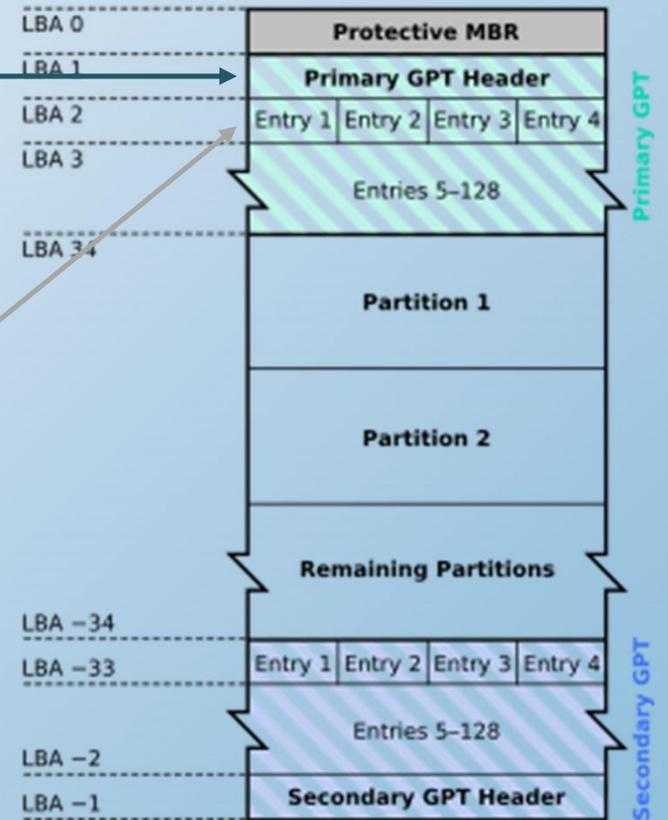
GUID Partition Table Scheme



5 Organisation des mémoires de masse

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00000001F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	55	AAU*
0000000200	45	46	49	2D	50	41	52	54	00	00	01	00	5C	00	00	00	EFI PART....\... Secteur 1
0000000210	FF	5F	FC	CD	00	00	00	00	01	00	00	00	00	00	00	00	ÿ úí.....
0000000220	AF	12	9E	3B	00	00	00	00	22	00	00	00	00	00	00	00	ŷ;....."
0000000230	8E	12	9E	3B	00	00	00	00	7D	A7	B4	9D	8E	41	47	49	Ž.Ž;....}S'.ŽAGI
0000000240	91	47	BC	3C	14	EC	3B	32	02	00	00	00	00	00	00	00	`G<.i;2.....
0000000250	80	00	00	00	80	00	00	00	CB	78	D9	E3	00	00	00	00	€...€...ËxÜä....
0000000260	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000270	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000280	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000290	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000002A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000002B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000002C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000002D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000002E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000002F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000300	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000310	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000320	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000330	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000340	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000350	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000360	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000370	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000380	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000390	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000003A0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000003B0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000003C0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000003D0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000003E0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00000003F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000400	28	73	2A	C1	1F	F8	D2	11	BA	4B	00	A0	C9	3E	C9	3B	(s*Ä.øð.*K. É>É; Secteur 2
0000000410	16	8C	5D	0E	32	3C	D5	47	A8	6C	FF	DA	7F	C0	F9	29	.(E].z<ÖG"lyÚ.Àù)
0000000420	00	08	00	00	00	00	00	00	FF	47	0B	00	00	00	00	00ÿG.....
0000000430	00	00	00	00	00	00	00	80	45	00	46	00	49	00	20	00€E.F.I. .
0000000440	73	00	79	00	73	00	74	00	65	00	6D	00	20	00	70	00	s.y.s.t.e.m. .p.
0000000450	61	00	72	00	74	00	69	00	74	00	69	00	6F	00	6E	00	a.r.t.i.t.i.o.n.
0000000460	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000470	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000480	16	E3	C9	E3	5C	0B	B8	4D	81	7D	F9	2D	F0	02	15	AE	.ããã\,M.)ù-8..@
0000000490	9C	AD	2A	FD	98	21	58	4A	B3	20	D1	35	E5	CE	A1	9C	œ.*ý"!XJ² Ñ5ãî;œ
00000004A0	00	48	0B	00	00	00	00	00	FF	47	0F	00	00	00	00	00	.H.....ÿG.....
00000004B0	00	00	00	00	00	00	00	80	4D	00	69	00	63	00	72	00€M.i.c.r.
00000004C0	6F	00	73	00	6F	00	66	00	74	00	20	00	72	00	65	00	o.s.o.f.t. .r.e.
00000004D0	73	00	65	00	72	00	76	00	65	00	64	00	20	00	70	00	s.e.r.v.e.d. .p.
00000004E0	61	00	72	00	74	00	69	00	74	00	69	00	6F	00	6E	00	a.r.t.i.t.i.o.n.
00000004F0	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0000000500	A2	A0	D0	EB	E5	B9	33	44	87	C0	68	B6	B7	26	99	C7	œ ðãã²3D+Ähç .g²MÇ
0000000510	07	B0	9D	EA	9C	22	09	4D	81	B6	34	A9	BE	19	8E	5E	.°.êœ".M.¶4@%.Ž^
0000000520	00	48	0F	00	00	00	00	00	FF	77	D2	1C	00	00	00	00	.H.....ÿwÖ.....
0000000530	00	00	00	00	00	00	00	42	00	61	00	73	00	69	00	00B.a.s.i.
0000000540	63	00	20	00	64	00	61	00	74	00	61	00	20	00	70	00	c. .d.a.t.a. .p.
0000000550	61	00	72	00	74	00	69	00	74	00	69	00	6F	00	6E	00	a.r.t.i.t.i.o.n.

GUID Partition Table Scheme



5. Organisation des mémoires de masse

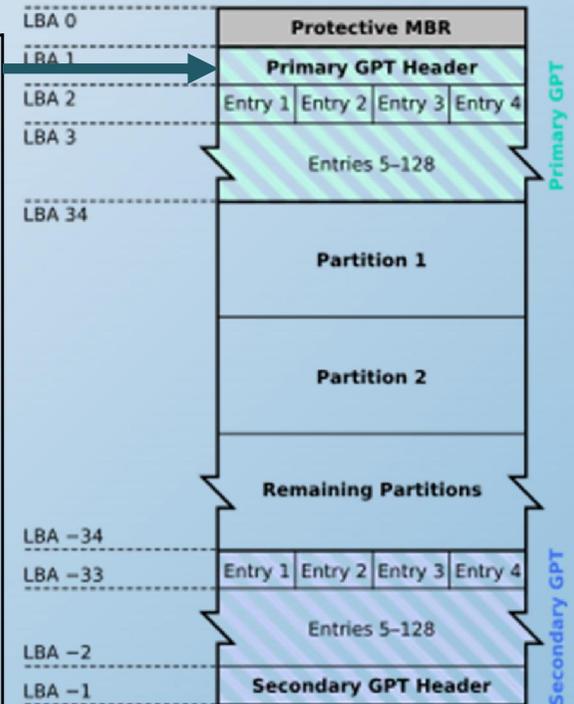


Le disque dur

Format de l'entête GPT (GPT Header)

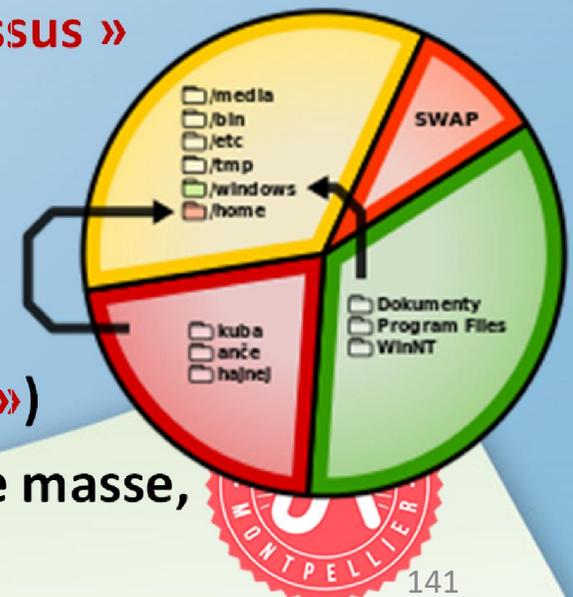
Offset	Longueur	Contenu
0	8 octets	Signature ("EFI PART", 45 46 49 20 50 41 52 54)
8	4 octets	Version (Pour la version 1.0, la valeur est 00 00 01 00, 1,0 en virgule fixe 16:16)
12	4 octets	Taille de l'entête (en octets, habituellement 5C 00 00 00 signifiant 92 octets)
16	4 octets	CRC32 de l'entête (de 0 à 92 octets - la taille de l'entête), avec ce champ à zéro pendant le calcul du crc32.
20	4 octets	Réservé, doit être initialisé à zero
24	8 octets	Adresse de cet entête GPT (emplacement de cet entête ; devrait être 1, le deuxième secteur du disque, pour l'entête primaire)
32	8 octets	Adresse de l'entête de l'autre GPT (ou backup ; devrait être le dernier secteur du disque pour l'entête primaire, et 1 pour l'entête secondaire)
40	8 octets	Première adresse utilisable pour les partitions
48	8 octets	Dernière adresse utilisable pour les partitions (à l'adresse suivante se trouve normalement la table de partitionnement secondaire)
56	16 octets	GUID du disque (aussi désigné comme UUID sur les UNIX)
72	8 octets	Adresse du tableau des partitions (normalement 2 pour le GPT primaire)
80	4 octets	Nombre de descripteurs de partition
84	4 octets	Taille en octets d'un descripteur de partition (habituellement 128)
88	4 octets	CRC32 de la table des partitions
92	*	réservé, la fin du bloc devrait être initialisée à zéro (420 octets sur un bloc de 512 octets)

GUID Partition Table Scheme



Le disque dur

- **La mémoire virtuelle (« swap memory »)**
- Principe : **utiliser une mémoire de masse dans le but** (entre autres) **de faire tourner plus de programmes que la RAM ne peut contenir**
→ empêche la saturation de la RAM et le blocage du système
- Le swap **permet** :
 - **d'augmenter le nombre de programmes qui s'exécutent en même temps** (« taux de multi-programmation ») → multi-tâches
 - de mettre en place des **mécanismes de protection de la mémoire**
 - de **partager la mémoire entre les différents « processus »** (= programmes en cours d'exécution)
- Sur un disque dur, on peut définir des partitions qui ne servent que comme mémoire virtuelle (**partitions de type « swap »**)
- **Inconvénient** : comme le « swap » est sur une mémoire de masse, **l'accès est beaucoup plus lent que pour la RAM...**

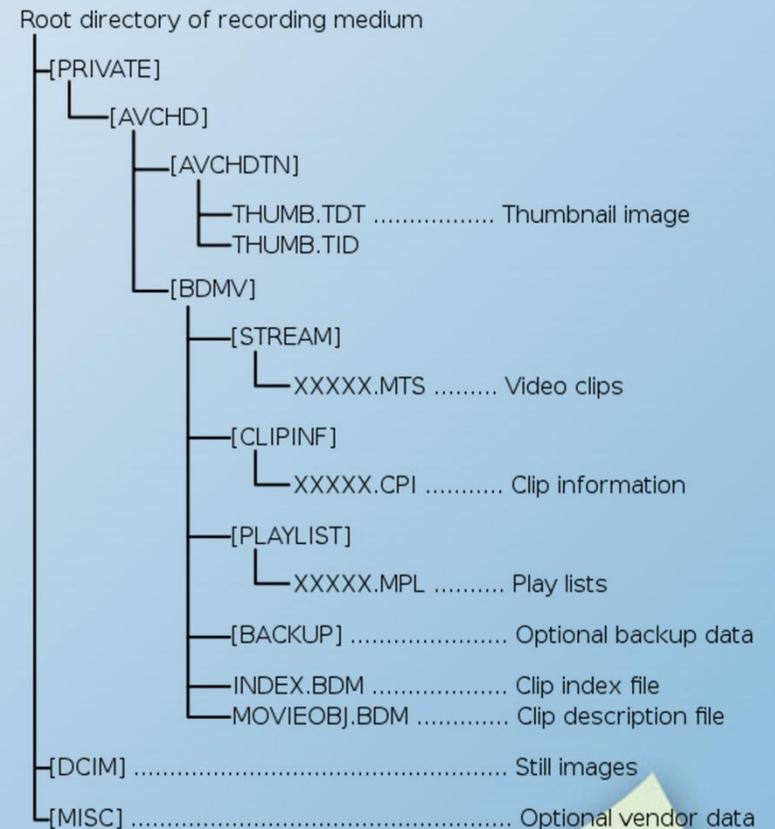


- **Les systèmes de fichiers (« File Systems » - FS)**
- Manière de stocker et organiser les informations sur les mémoires de masse, **permettant de :**
 - **traiter** les données
 - **conserver** les données
 - **partager** les données entre **plusieurs programmes**
- Autres manières d'organiser les données : bases de données, fichiers indexés = fichiers qui contiennent un *index* (clé) – valeur qui se trie par ordre croissant ou décroissant
- Dans les FS, les fichiers sont stockés dans les « clusters »

Le disque dur

- **Les systèmes de fichiers (« File Systems » - FS)**

- De point de vue de l'utilisateur, un **FS** est vu comme une **ARBORESCENCE de REPERTOIRES** qui contiennent soit des fichiers, soit d'autres répertoires = **ARBRE**
- L'arborescence est parcourue de façon **RECURSIVE**
- Chaque **fichier** est décrit par des **métadonnées** (« données concernant d'autres données » = informations sur le fichier) **et par les données qu'il contient lui-même**

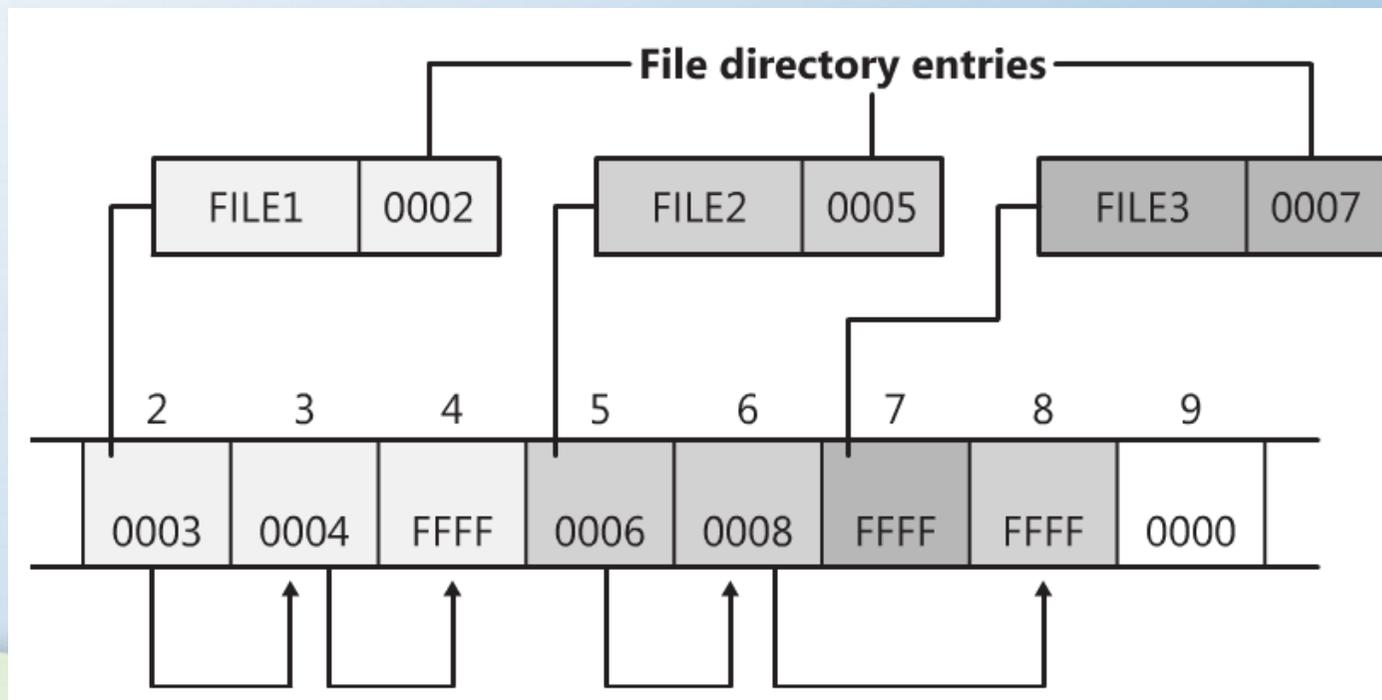


Le disque dur

- **Les systèmes FAT (« File Allocation Table »)**
- Développés par Microsoft depuis 1977
- 3 variantes : FAT 12 (utilise 12 bits pour adresser les clusters), FAT 16 (16 bits), FAT 32 (28 bits – les 4 bits de poids fort étant réservés)
- vFAT = extension optionnelle du FAT permettant de prendre en charge les noms de fichiers de + de 8 caractères (s'applique à toutes les versions de FAT)
- Utilisation : DOS, Windows, clés USB, cartes SD, appareils photo numériques, phase de démarrage des systèmes compatibles UEFI...
- **Caractéristique :**
chaque répertoire contient **une table**
qui associe les **noms de fichiers à leur taille**
+
un **index** pointant sur la FAT = zone disque indiquant,
pour chaque bloc, l'index **du bloc suivant du même fichier**

Le disque dur

- **Les systèmes FAT (« File Allocation Table »)**
- Chaque répertoire contient **une table** qui associe les **noms de fichiers à leur taille**
+
un **index** pointant sur la FAT = zone disque indiquant, pour chaque bloc, l'**index du bloc suivant du même fichier**



Le disque dur

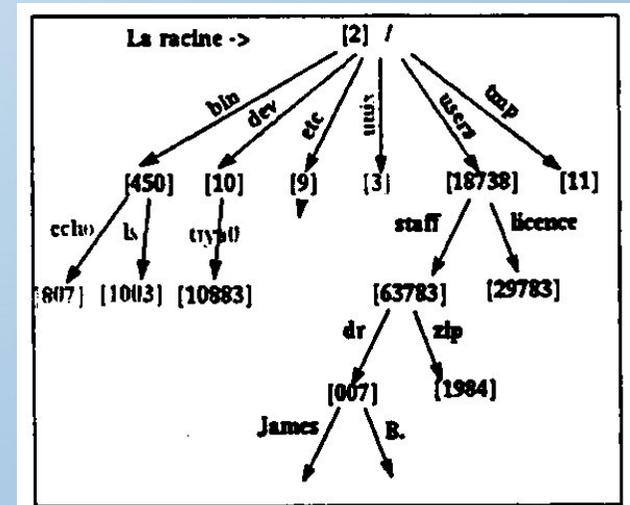
- **Les systèmes FAT (« File Allocation Table »)**
- Valeurs des index du FAT

FAT12	FAT16	FAT32	Description
0x000	0x0000	0x?0000000	<i>Cluster vide</i>
0x001	0x0001	0x?0000001	<i>Cluster réservé</i>
0x002 - 0xFEFF	0x0002 - 0xFFEF	0x?0000002 - 0x?FFFFFFEF	<i>Cluster utilisé, pointeur vers le cluster suivant du fichier</i>
0xFF0 - 0xFF6	0xFFFF0 - 0xFFFF6	0x?FFFFFFF0 - 0x?FFFFFFF6	Valeurs réservées
0xFF7	0xFFFF7	0x?FFFFFFF7	« Mauvais cluster »
0xFF8 - 0xFFFF	0xFFFF8 - 0xFFFFF	0x?FFFFFFF8 - 0x?FFFFFFF	<i>Cluster utilisé, dernier cluster d'un fichier</i>

Le disque dur

• Les systèmes UNIX (« EXT »)

- **Caractéristique** : les fichiers et répertoires sont identifiés par le numéro d'**inode**
- **i-node (« Index node »)** = structure de données regroupant toutes les **informations sur un fichier** (protections, dates, moyens de retrouver le contenu), **sauf leur nom**
- **Le nom de chaque fichier est stocké dans le répertoire, associé au numéro d'inode**
- **Un répertoire est simplement un fichier dont les blocs de données contiennent une liste de couples (nom, numéro d'inode) qui pointent sur les fichiers censés être contenus dans le répertoire**
- **La racine du système de fichiers (/) correspond à l'inode 2**
- **Exemple : répertoire contenant 3 fichiers : a.txt, b.sh, c.pl → il sera composé de 5 couples : {(. , 43), (.. , 234), (a.txt, 36), (b.sh, 245), (c.pl, 24524)}**
- **Un répertoire contient des entrées de 16 octets : 2 octets pour l'inode et 14 pour le nom du fichier → $2^{16} = 65536$ numéros d'inodes différents = nb. max de fichiers**



Le disque dur

- Les systèmes UNIX (« EXT »)

- Avantage du fait que l'inode ne contient pas le nom du fichier :
possibilité de donner plusieurs noms à un même fichier = **liens**
- Lien = couple (nom de fichier, numéro d'inode)
- On peut créer 2 type de liens :

Physiques :

- créer un lien physique = créer un **nouveau nom pour le même inode**
- il est donc impossible de faire un lien physique vers un répertoire :
le système de fichiers doit rester arborescent
- impossible de faire un lien physique vers un autre volume (partition)

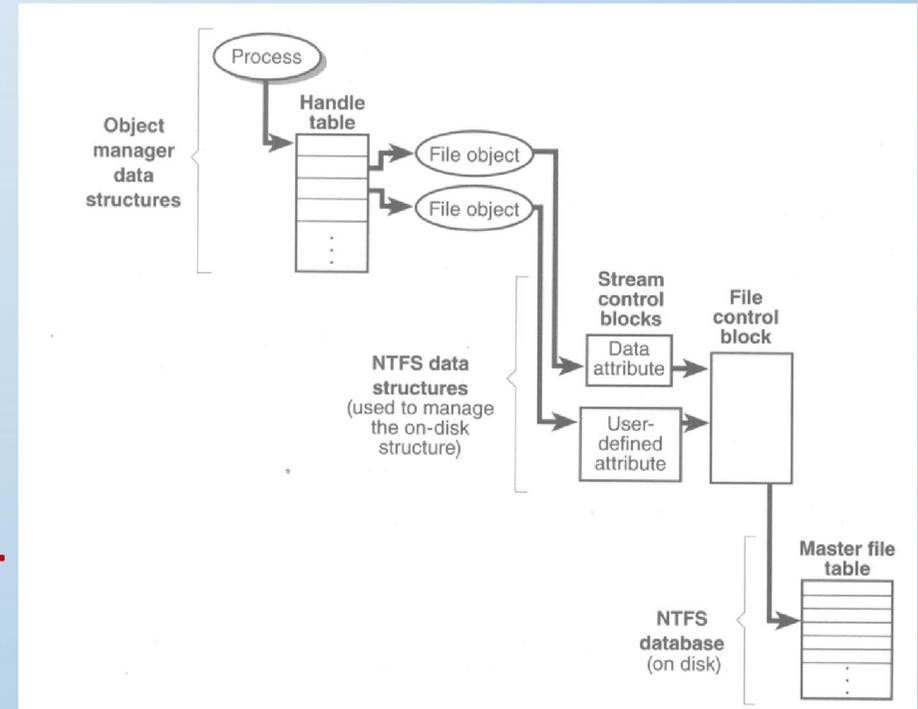
Symboliques (raccourcis) :

- un lien symbolique permet de **donner un autre nom au fichier**,
mais **n'utilise pas l'inode** physique du fichier
- le bloc de données contient une chaîne de caractères
représentant le chemin (relatif ou absolu) vers le fichier
- il est donc possible de créer des liens symboliques vers des répertoires
- on peut faire un lien symbolique vers un autre volume (partition)

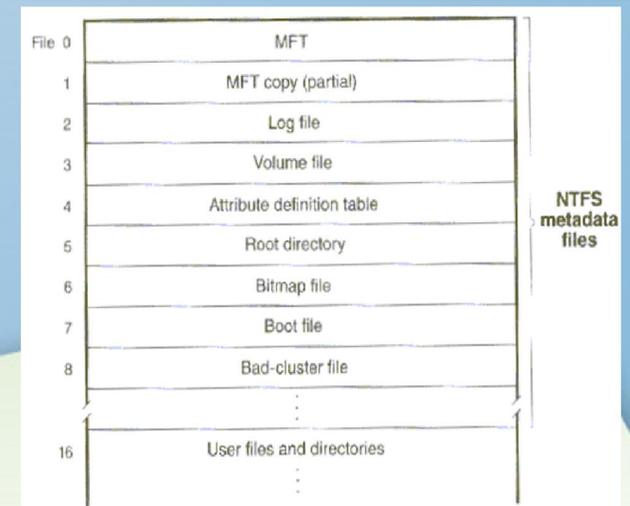
Le disque dur

- **Le système NTFS (« New Technology File System »)**

- Développé par Microsoft pour permettre à ses systèmes d'exploitation (Windows NT, XP, 7, 8, 10) de donner des droits d'accès aux fichiers (ce que le FAT ne permet pas) et d'assurer le multi-tâche
- Structure similaire à une base de données
- Il utilise une table d'allocation appelée MFT (« Master File Table »)



- La MFT occupe plusieurs milliers de clusters (taille variable)
- Par défaut, NTFS lui réserve 12,5 % de l'espace disponible sur la partition (les données ne pouvant pas être écrites sur cette zone réservée), ce qui évite sa fragmentation



Le disque dur

- **Fragmentation = éparpillement du contenu d'un fichier dans des clusters non contigus**
→ **augmentation du temps d'accès au contenu du fichier**
 - **UNIX :**
 - Les fichiers sont stockés à des multiples endroits sur le disque, en laissant des espaces énormes entre eux → les fichiers peuvent "grossir" sans obstruction
 - Si le disque est encombré et qu'un fichier a besoin de plus de place pour grossir tout en restant contigu, le système essaye de le réécrire entièrement à un endroit où il y a de la place :
 - Le noyau calcule le nombre de blocs nécessaires au stockage de chaque fichier sur le disque dur
 - Si un tel nombre de clusters libres contigus est trouvé sur le disque, il stocke le fichier sur ces clusters, contigus, et le fichier n'est pas fragmenté
 - S'il ne trouve pas assez de clusters libres contigus, il scinde le fichier en plusieurs groupes de clusters et éparpille ces groupes sur le disque dur : il tente de minimiser le nombre de groupes et par conséquent remplit les plus grands espaces de clusters vides contigus en premier
- **dans les systèmes Unix, la fragmentation est faible et fait suite à un espace disque trop faible**
- **la défragmentation est rarement nécessaire**

Le disque dur

- **Fragmentation** = éparpillement du contenu d'un fichier dans des clusters non contigus
→ augmentation du temps d'accès au contenu du fichier
 - **Microsoft :**
 - Les fichiers sont écrits consécutivement l'un après l'autre
 - Le noyau NT essaie de combler les trous dans le sens où il fragmente le fichier pour ne pas laisser au début du disque des zones avec des clusters libres
 - Dans les systèmes Microsoft, la fragmentation est due aux multiples suppressions, modifications, copies de fichiers... → zones de clusters libres → fragmentation
- **fragmentation fréquente sur les systèmes Windows** car il est courant pour un utilisateur de manipuler des fichiers
- **nécessité de défragmenter** assez régulièrement si l'on souhaite retrouver des vitesses de lecture élevées

Le disque dur : SSD vs. HDD

Caractéristique	SSD	Disque dur (HDD)
Temps d'accès aléatoire	Environ 0,1 ms	De 2,9 à 12ms
Vitesse de lecture/écriture	De 27 Mo/s à 3 Go/s	De 12 à 260 Mo/s
IOPS	De 8 000 à 3 000 000 (connexion PCIe, plusieurs téraoctets)	Dépend de la vitesse de rotation et du nombre d'axes
Fragmentation	Aucun effet	Dépend du système de fichiers
Bruit	Très peu voire aucun	Variable mais ayant tendance à s'accroître avec le temps et selon la charge d'utilisation
Vulnérabilités	Sensible au nombre de cycles d'écriture Coupures de courant qui peuvent rendre le lecteur irrécupérable sur certains (anciens) modèles	Chocs et vibrations, sensibles aux champs magnétiques
Taille	4,57-6,35 cm (1,8-2,5") (en fonction des modèles)	4,57-6,35-8,89 cm (1,8-2,5-3,5") (en fonction des modèles)
Masse	Quelques dizaines de grammes	Jusqu'à près de 700 g
Durée de vie	Garantie constructeur allant de 1 à 10 ans Cycles d'écriture garantis : 10 000 (SLC), 5 000 (MLC) et 1000 (TLC)	2 à 5 ans (voire plus) en fonction de la charge et des conditions d'utilisation
Rapport coût-capacité	environ 0,4 €/Gio (2014)	environ 0,04 €/Gio (2014)
Capacité de stockage	Jusqu'à 16 To	Jusqu'à 10 To.
Consommation	0,1 - 0,9 W (veille) jusqu'à 0,9 W (activité)	0,5 à 1,3 W (veille) 2 à 4 W (activité)

Le disque dur : **SSD vs. HDD**

- perfs en lecture **SSD** >> HDD, mais leur conception fait qu'ils ne peuvent réécrire que des zones bien plus grandes qu'un secteur disque
→ **perfs réelles en écriture bcp moins bonnes si partition relativement pleine**
- les HDD laissent beaucoup de temps libre au processeur entre deux opérations, contrairement aux **SSD** du fait de leur absence de délais mécaniques
→ **perte de réactivité possible faute de temps processeur à consacrer au clavier**
- Durée de vie pouvant être bcp + réduite SSD si écriture fréquente
- Prix SSD > HDD

• **Type d'adressage ?**