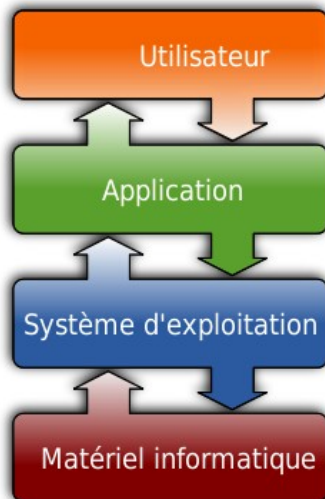


Mod02 – Architecture et fonctions d'un système d'exploitation

1. Définition et rôle



Le système d'exploitation, abrégé SE (en anglais operating system, abrégé OS), est l'ensemble de programmes d'un ordinateur qui sert d'interface entre le matériel et les logiciels applicatifs.

On pense à rajouter la couche des drivers entre l'OS et le matériel.

Les fonctions d'un système d'exploitation est de :

1. Contrôler et partager des ressources physiques (CPU, la mémoire, les périphériques, les entrées / sorties), et des ressources **logiques** (fichiers)
2. Gérer une ou des interfaces Homme-machine (IHM - texte ou graphique).
3. Gérer les communications : entre les machines et/ou entre les différents programmes.
4. Assurer la sécurité.
5. Assurer le parallélisme : la gestion multiprocesseurs; multi-utilisateurs, etc.

Un système d'exploitation est composé :

- D'un noyau ;
- D'une interface utilisateur (IHM)
- D'un système de fichiers
- D'un ensemble d'outils système ;
- De programmes applicatifs de base.

a) Le noyau

Le noyau (ou cœur, **Kernel**) assure les fonctionnalités suivantes :

- Gestion des périphériques (au moyen de pilotes) ;
- Gestion de l'exécution des programmes (aussi nommé processus) :
 - Gestion de la mémoire attribuée à chaque processus ;
 - Ordonnancement des processus (répartition du temps d'exécution sur le ou les processeurs)
 - Synchronisation et communication entre processus (services de synchronisation, d'échange de messages, mise en commun de segments de mémoire, etc.)

Il s'agit de la couche primordiale, celle qui est lancée lors du démarrage de l'ordinateur que l'on appelle couramment le **boot**. Grâce à celui-ci, les premiers services peuvent accéder aux applications système : gestion de la mémoire, accès aux disques durs et accès aux périphériques. Il gère donc les ressources de l'ordinateur et permet aux différents composants matériels et logiciels de communiquer entre eux.

b) L'interface utilisateur (IHM)

Elle permet à l'homme de se servir de l'ordinateur.

- **La commande en ligne**

La ligne de commande (en anglais CLI pour Command Line Interface) était la seule interface disponible sur les ordinateurs des années 1970. Elle est encore utilisée en raison de sa puissance, de sa grande rapidité et du peu de ressources nécessaires à son fonctionnement.

- **L'interface graphique**

L'interface graphique (en anglais GUI pour Graphical User Interface) s'oppose à l'interface en ligne de commande. Elle répond à la norme WIMP (Window Icon Menu Pointer). A destination du grand public pour pouvoir être plus convivial, simple d'utilisation.

c) Le système de fichiers

Un système de fichiers (**FS** ou filesystem en anglais) ou système de gestion de fichiers (**SGF**) est une structure de données permettant de **stocker les informations** et de les **organiser** dans des fichiers sur ce que l'on appelle des mémoires secondaires (disque dur, clé USB, etc.).

Ce stockage de l'information est **persistant**. Une telle gestion des fichiers permet de traiter et de conserver des quantités importantes de données ainsi que de les partager entre plusieurs programmes informatiques. Il offre à l'utilisateur une vue abstraite sur ses données et permet de les localiser à partir d'un **chemin** d'accès qui peut être **absolu** ou **relatif**.

d) Les outils système

Les outils système permettent de configurer le système (gestion des comptes des utilisateurs, configuration des paramètres réseau, démarrage automatique des services, etc.) :

- panneau de configuration
- fichiers de configuration

e) Les programmes applicatifs de base

Les programmes applicatifs de base offrent des services à l'utilisateur (calculatrice, éditeur de texte, navigateur web, etc.). Ces programmes applicatifs sont souvent fournis en bundle (en vente liée) avec le système d'exploitation. Certaines personnes estiment qu'ils ne font pas réellement partie du système d'exploitation. La séparation entre les programmes applicatifs de base et le système d'exploitation est difficile à définir, du fait que l'un devient inutile sans l'autre, et que bon nombre d'applications sont programmées en supposant que les programmes applicatifs de base sont toujours présents.

2. Installation et partitionnement

Un système d'exploitation ne s'installe pas sur un disque dur. **Il s'installe sur une partition**. D'où la nécessité de partitionner le disque dur d'un ordinateur. Cela sert par exemple à **installer des systèmes d'exploitation différents n'utilisant pas le même système de fichiers**. Il y aura donc au minimum autant de partitions que de systèmes d'exploitation utilisant des systèmes de fichiers différents.

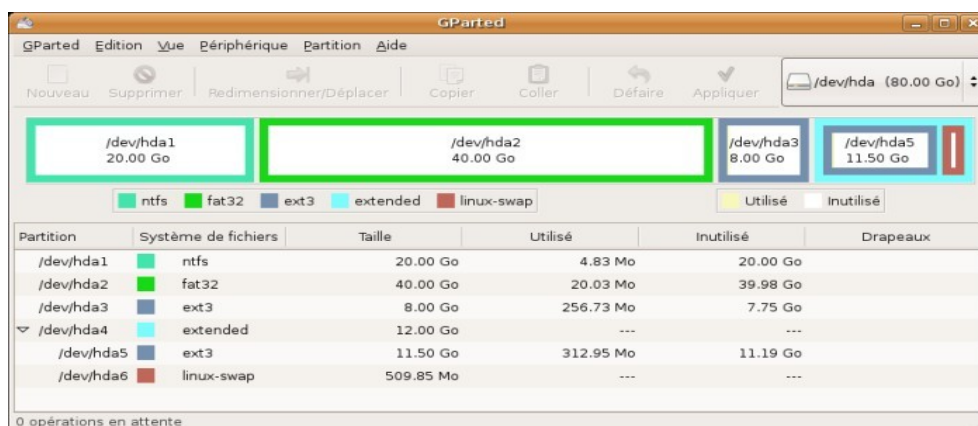
Dans le cas d'un utilisateur d'un système d'exploitation unique, une seule partition de la taille du disque peut suffire, sauf si l'utilisateur désire en créer plusieurs pour faire par exemple plusieurs **lecteurs** dont **les données** sont séparées.

Historiquement, la manière de partitionner les supports de stockage était d'avoir **3 sortes de partitions** :

- la partition **principale**,
- la partition **étendue**,
- les lecteurs **logiques** (ou partitions logiques)

Un disque peut contenir **jusqu'à quatre partitions principales, ou trois partitions principales et une partition étendue**. Dans la partition étendue l'utilisateur peut créer des **lecteurs logiques** (c'est-à-dire "simuler" plusieurs disques durs de taille moindre).

Cette **méthode perdure encore** de nos jours et c'est celle que nous utilisons sous Ubuntu (ou Debian).



Cependant, les limites de la **table de partitions** située à l'intérieur du **MBR** (qui contient aussi le chargeur de démarrage comme **GRUB**) et le remplacement des **BIOS** des ordinateurs personnels par l'**EFI** (ou l'**UEFI**) devrait achever la transition vers le mode de partitionnement **GPT**.

On appelle **partitionnement** le processus qui consiste à écrire les secteurs qui constitueront la **table de partition** (qui contient les informations sur la partition) :

- taille de celle-ci en terme de nombre de **blocs** (ou de **secteurs**),
- **position** par rapport à la partition principale,
- **types** de partitions présentes,
- **systèmes d'exploitation** installés.

Lorsque la partition est créée, on la localise par son **adressage matériel** (ex : /dev/sda1), ou on lui donne un **nom de volume**, ou encore un **UUID** est généré au formatage pour permettre de l'identifier facilement.

3. MBR (Master Boot Record)

Le **secteur de démarrage** (appelé Master Boot Record ou MBR en anglais) est le premier secteur d'un disque dur (cylindre 0, tête 0 et secteur 1), il contient la **table de partition principale** (en anglais partition table) et le code, appelé **boot loader**, qui, une fois chargé en mémoire, va permettre d'amorcer (booter) le système.

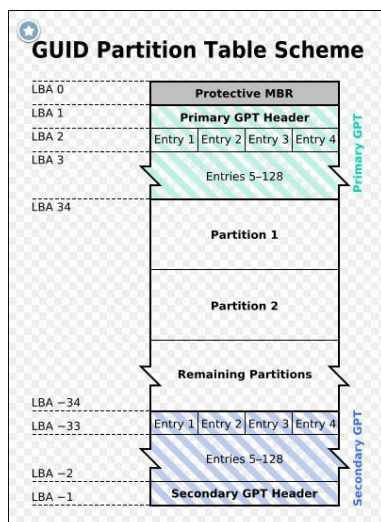
Ce programme, une fois en mémoire, va déterminer sur quelle partition le système va s'amorcer, et il va démarrer le programme (appelé **bootstrap**) qui va amorcer le système d'exploitation présent sur cette partition.

D'autre part, c'est ce secteur du disque qui contient toutes les informations relatives au disque dur (fabricant, numéro de série, nombre d'octets par secteur, nombre de secteurs par cluster, nombre de secteurs,...). Ce secteur est donc le secteur le plus important du disque dur, il sert au setup du **BIOS** à reconnaître le disque dur. Ainsi, sans celui-ci votre disque dur est inutilisable, c'était pendant longtemps donc une cible de prédilection pour les virus.

4. GPT (GUID Partition Table)

Une **table de partitionnement GUID**, en anglais **GUID Partition Table (GPT)** est un standard pour décrire la table de partitionnement d'un disque dur. Bien qu'il fasse partie du standard **EFI (Extensible Firmware Interface)**, qu'Intel propose en remplacement du PC BIOS, il est aussi utilisé sur certains BIOS à cause des limitations de la table de partitionnement du MBR qui limite la taille des partitions à 2,2 To. GPT gère les disques durs et partitions jusqu'à 9,4 Zo.

La prise en charge de GPT parmi les principaux systèmes reste encore limitée.



Unités d'octets v · d · m						
Ordre de grandeur	Système international (SI)			Préfixes binaires		
	Unité	Notation	Valeur	Unité	Notation	Valeur
1	octet	o	1 octet	octet	o	1 octet
10 ³	kiloctet	ko	10 ³ octets	kibioctet	Kio	2 ¹⁰ octets
10 ⁶	mégaoctet	Mo	10 ⁶ octets	mébioctet	Mio	2 ²⁰ octets
10 ⁹	gigaoctet	Go	10 ⁹ octets	gibioctet	Gio	2 ³⁰ octets
10 ¹²	téraoctet	To	10 ¹² octets	tébioctet	Tio	2 ⁴⁰ octets
10 ¹⁵	pétaoctet	Po	10 ¹⁵ octets	pébioctet	Pio	2 ⁵⁰ octets
10 ¹⁸	exaoctet	Eo	10 ¹⁸ octets	exbioctet	Eio	2 ⁶⁰ octets
10 ²¹	zettaoctet	Zo	10 ²¹ octets	zébioctet	Zio	2 ⁷⁰ octets
10 ²⁴	yottaoctet	Yo	10 ²⁴ octets	yobioctet	Yio	2 ⁸⁰ octets

5. Système de fichiers

Système de gestion de fichiers (SGF) ou File System (FS) : c'est une façon de **stocker** les informations et de les **organiser** dans des fichiers sur les mémoires secondaires (disque dur, clé USB, etc.).

Une telle gestion des fichiers permet de traiter, de conserver des quantités importantes de données ainsi que de les partager entre plusieurs programmes informatiques. Il offre à l'utilisateur une vue abstraite sur ses données et permet de les localiser à partir d'un **chemin d'accès et une arborescence**.

Le système qui va permettre à un système d'exploitation d'utiliser l'espace disque pour stocker et utiliser des fichiers. Historiquement, le système de fichiers est basé sur la gestion des **clusters** (en français « **unité d'allocation** »), c'est-à-dire la plus petite unité de disque que le système d'exploitation est capable de gérer.

Un cluster est constitué d'un ou plusieurs secteurs, ainsi plus la taille d'un cluster est importante, moins le système d'exploitation aura d'entités à gérer...

En contrepartie, étant donné qu'un système d'exploitation ne sait gérer que des unités d'allocation entière, c'est-à-dire qu'un fichier occupe un nombre entier de cluster, le gaspillage est d'autant plus grand qu'il y a de secteurs par cluster. On comprend alors toute l'importance du choix du système de fichiers.

Au fur et à mesure du temps, les systèmes de fichiers ont évolué avec les systèmes d'exploitation : FAT16, FAT32, NTFS, Ext3, Ext4, ReiserFS, JFS, Linux Swap, MFS (Macintosh File System), ExFAT, etc. et ne sont pas toujours bien **compatibles** entre eux !

5. Le marché actuel

Destinés aux **téléphones mobiles**, on trouve les 2 principaux iOS de Apple's et Android de Google.

Destinés aux **ordinateurs de bureau**, on trouve Windows 10 de Microsoft, MacOS de Apple, des distributions Linux (Ubuntu étant la plus utilisées), et parfois Android de Google.

Destinés aux **serveurs**, on trouve des dérivés de UNIX, des distributions de Linux (Debian et Suse étant les plus utilisées), Windows Server de Microsoft.

Sur les **box**, les **NAS** et les **GPS**, on trouve généralement des distributions de Linux légères (souvent sans interface graphique).

6. Licence propriétaire / Licence libre

Format ouvert : format pour lequel le créateur publie les spécifications, permettant ainsi à d'autres sociétés de réutiliser ce format. Le créateur se réserve toutefois le droit de demander des redevances. Par exemple pour le format PDF d'Acrobat : Acrobat publie les spécifications du format PDF, donnant ainsi à quiconque la possibilité de créer une application qui exploite le format. Toutefois, Acrobat ne demande une redevance qu'à quelques très importantes sociétés commerciales.

Format, logiciel libre (en anglais : **open-source**) : le format est ouvert (pour un logiciel, cela implique que son code source peut être téléchargé sur Internet), mais il n'appartient à personne, donc personne ne peut prétendre à des droits financiers sur le produit. Par conséquent, n'importe qui peut le modifier, l'adapter, et utiliser commercialement le produit. C'est le cas de Linux qui existe dans de nombreuses versions différentes, souvent gratuites, parfois payantes. Il existe différentes versions de licences libre : (GPL, license BSD, etc.).

Précision : quelle différence entre le [code source](#) et le [code compilé](#) ? Exemples : Mac OS X, Linux, The GIMP, OpenOffice, format PNG, format OGG Vorbis, etc.

Format, logiciel propriétaire : par exemple Microsoft Windows, . Le logiciel propriétaire est soumis au droit d'auteur. Le propriétaire décide de la façon dont est utilisé son logiciel, et de la rétribution associée. "Propriétaire" n'est pas forcément "payant".

Gratuiticiel (le terme anglais **freeware** est plus répandu) : le gratuiticiel est un logiciel propriétaire, distribué gratuitement. Il n'est pas libre, donc on n'a pas le droit de consulter son code source ni le modifier.

Partagiciel (le terme anglais **shareware** est plus répandu) : le partagiciel est un logiciel propriétaire, donc soumis au droit d'auteur, mais l'auteur autorise à ce qu'on l'utilise gratuitement pendant une certaine période, ou un certain nombre d'utilisations. La frontière est étroite entre **partagiciel** et **version de démonstration**.

7. Qu'est-ce qu'une distribution Linux ?

Linux est le système d'exploitation le plus répandu au monde (sauf pour les ordinateurs de bureau).

Une distribution Linux est composée :

- d'un **noyau Linux**
- éventuellement d'une **interface graphique**, les plus répandues étant **KDE** et **Gnome**,
- d'une **suite logicielle** adaptée à la finalité de la distribution : outils bureautiques, graphiques, serveurs, sécurité, etc.

Une **distribution live** d'un système d'exploitation est une distribution qui se lance **sans nécessiter d'installation préalable sur un disque dur**. L'ordinateur boote sur un CD ou une clé USB qui va lancer directement le système d'exploitation. Un disque dur n'est même pas nécessaire. Bien qu'une mémoire de masse (disque dur ou clé USB) ne soit pas nécessaire, c'est toutefois pratique pour pouvoir paramétrer le système.

9. Sources

- Le cours de Pierre Dejeans
- [http://www.bts-sio.com/cours/SI1/3.%20Introduction%20aux%20SE%20\(2013\)/Systemes%20d%27exploitation.php](http://www.bts-sio.com/cours/SI1/3.%20Introduction%20aux%20SE%20(2013)/Systemes%20d%27exploitation.php)