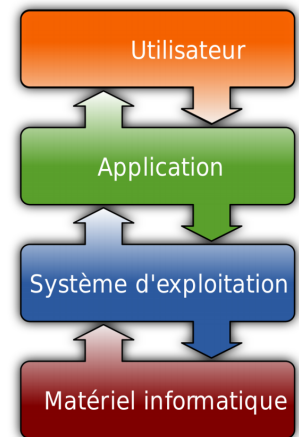


Systèmes d'exploitation

Qu'est-ce qu'un système d'exploitation ?

Un système d'exploitation (*Operating System* en anglais, abrégé OS) est un ensemble de programmes. Il a plusieurs fonctions :

- c'est un intermédiaire entre l'utilisateur et les applications qu'il utilise (qui sont aussi des programmes) et le matériel;
- c'est lui qui partage les *ressources* entre les différents programmes en train d'être exécutés voire entre les différents utilisateurs;
- c'est lui qui protège la machine des applications, et les applications les unes des autres;
- c'est lui qui gère l'adaptation entre l'*interface* (graphique par exemple) et le matériel.



Les ressources, ce sont entre autres :

- le matériel qui compose l'ordinateur proprement dit (CPU, mémoire, carte graphique...) et les *périphériques* (imprimante, clé usb, clavier);
- les fichiers;
- les éventuelles connexions à des réseaux.



Logos de divers OS.

Il existe de multiples OS.

- pour PC : WINDOWS et beaucoup de distributions de LINUX;
- pour APPLE : MACOS;
- pour iPhone et iPad : IOS;
- pour beaucoup de smartphones : ANDROID;

Les appareils connectés ainsi que les « box » des fournisseurs d'accès à Internet et les consoles de jeux disposent aussi de leurs OS. Certains OS sont *gratuits*, d'autres *payants*.

Le système de gestion des fichiers

Un *fichier* est un ensemble de données numériques réunies sous un même nom et enregistré sur un support de mémoire permanent appelé *mémoire de masse* (ce peut être un disque dur, un DVD-rom, une clé USB, une carte SD).

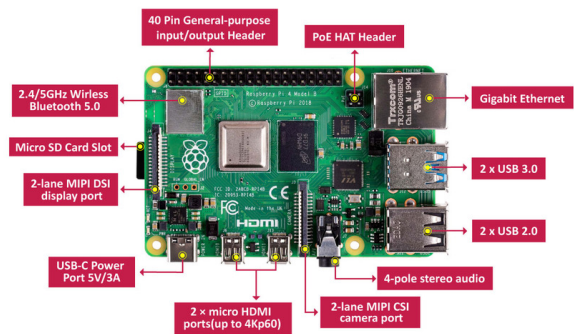
Un *système de gestion de fichiers* est une façon de stocker les fichiers sur la mémoire de masse. La plupart des systèmes d'exploitation stockent les fichiers dans des *répertoires* organisés de manière hiérarchique selon une *arborescence* (voir plus bas).

L'exemple de Linux et du shell

En 1970 naît UNIX, l'un des premiers système d'exploitation multi-utilisateurs. Pour connaître son histoire, [visiter ce lien](#).

Cependant ce système d'exploitation n'est pas *libre*, c'est à dire que son code source n'est pas à la disposition du public.

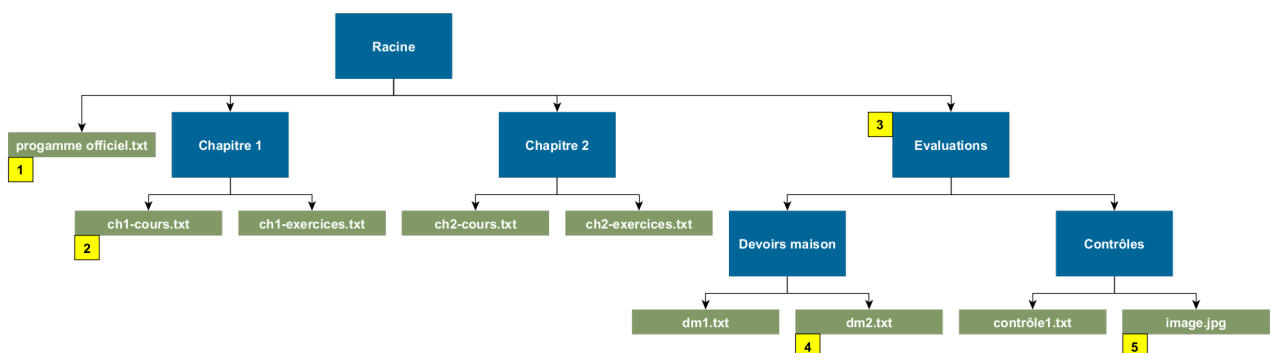
En 1991, un étudiant finlandais du nom de Linus Torvalds entreprend de créer un clone d'UNIX en le réécrivant totalement.



Le Raspberry Pi 4, un ordinateur très modeste mais complet. Longueur : 10 cm. Prix : 50€

Ce nouveau système d'exploitation libre s'appelle LINUX et est désormais largement utilisé.

C'est la plupart du temps une version (appelée *distribution*) de LINUX que l'on installe sur les ordinateurs Raspberry Pi.



Exemple d'arborescence de fichiers.

Dans un système de fichier tel que celui de LINUX, la racine se note / et les séparateurs (équivalents des flèches du graphique ci-dessus) aussi.

Il y a deux manières d'accéder aux fichiers.

Chemin absolu

On part de la racine et on écrit le chemin pour arriver à l'objet désiré.

Ainsi le chemin absolu de l'objet 1 est `/programme officiel.txt` et celui de l'objet 5 est `/Evaluations/Contrôles/image.jpg`

Exercice 1

Donner les chemins absolus des autres objets.

Chemin relatif

On part d'un emplacement précis (à l'intérieur d'un répertoire) et l'on indique le chemin pour aller à l'objet voulu. S'il faut remonter d'un cran dans l'arborescence on utilise la notation `..` comme ceci :

- si l'on est dans le répertoire **Evaluations** et qu'on veut accéder à **dm2.txt** alors son chemin relatif est **Devoirs maison/dm2.txt**;
- si l'on est dans **Contrôles** et qu'on veut accéder à **ch1-exercices.txt** alors son chemin relatif est **../../Chapitre 1/ch1-exercices.txt** car il faut d'abord remonter l'arborescence de 2 crans.

Exercice 2

On est dans le répertoire **Contrôles**, donner les chemins relatifs des objets 1 à 5.

Le shell

Nous sommes habitués à un environnement graphique pour gérer les copies et déplacements de nos dossiers (et bien d'autres choses comme par exemple sélectionner des fichiers à compresser) mais ce n'est pas la seule manière. Dans tout système d'exploitation muni d'un environnement graphique, on peut trouver un *terminal* (aussi appelé *invite de commande*, ou *shell*) parce que

- c'est avec cette interface textuelle minimaliste que l'on interagissait avec l'OS avant;
- finalement quand on sait (très bien) s'en servir, on peut faire beaucoup plus de choses plus rapidement avec un shell et au clavier qu'avec un clavier, une souris et un environnement graphique.

Exercice 3

Faire le début de l'activité « utiliser le shell » de M. Quinson.

<http://katacoda.com/mquinson>