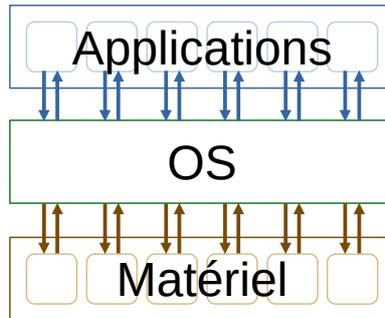


SYSTÈMES D'EXPLOITATIONS

1) Rôle d'un système d'exploitation

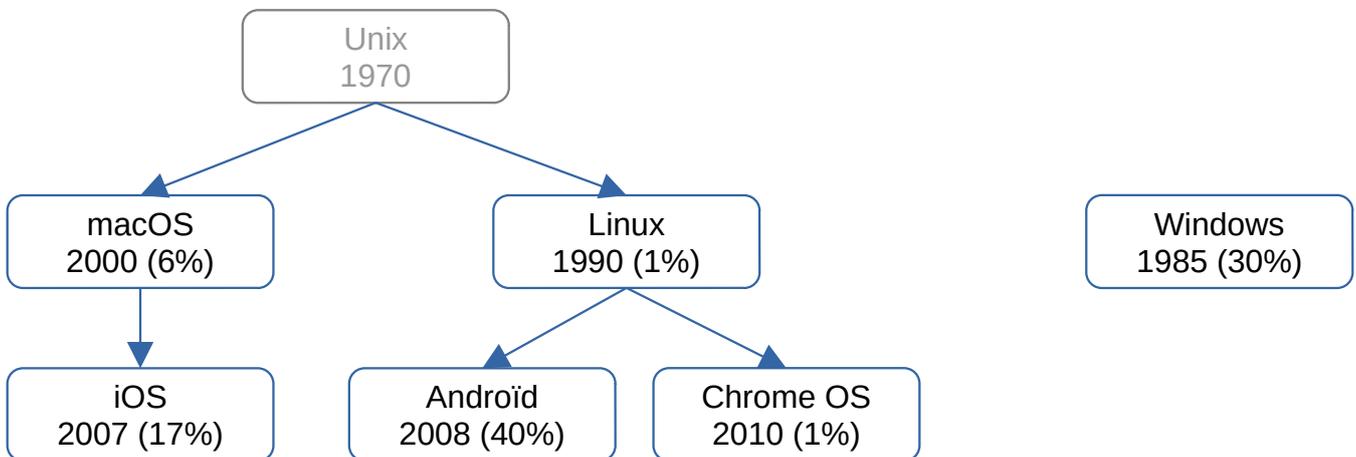
Dans les années 1950, un ordinateur ne pouvait exécuter qu'un seul programme à la fois et un programme devait être en partie réécrit pour pouvoir être exécuté sur un autre ordinateur. Assez vite, on a donc eu l'idée d'ajouter un logiciel intermédiaire qui serve d'interface entre les différents programmes à exécuter et l'ordinateur lui-même. C'est ainsi que sont nés les premiers systèmes d'exploitation (Operating System – OS).



De nos jours, un OS permet notamment :

- D'exécuter en parallèle plusieurs applications en même temps (multitâche)
- De gérer plusieurs utilisateurs (multi-utilisateur).
- De masquer les particularités matérielles de la machine sur laquelle il est installé en faisant appel à des concepts plus « abstraits » (par exemple la notion de fichier est une notion abstraite indépendante de la façon dont les données sont concrètement stockées sur le support physique).

Les principaux OS aujourd'hui :



(Statistiques d'après <https://gs.statcounter.com/os-market-share>)

L'OS est chargé au démarrage de l'ordinateur et ensuite tourne en permanence. Il est habituellement composé :

- D'un noyau (kernel)
- De bibliothèques (API) permettant aux applications d'accéder au matériel et aux services offerts par l'OS
- D'une interface graphique (GUI) et/ou d'une interface en ligne de commande (CLI)
- D'un ensemble d'outils systèmes

(Pour en savoir plus sur l'histoire d'Unix : <https://www.youtube.com/watch?v=Za6vGTLp-wg>)

II) Le noyau

Le noyau gère notamment :

- Les processus (process)
Un processus est un programme en cours d'exécution. Le noyau alloue à chaque processus une partie de la mémoire et une partie du temps d'utilisation du processeur. Il redirige les entrées-sorties vers les processus auxquels elles sont destinées...
- Le matériel (hardware)
La mémoire (cache, mémoire principale), le processeur (cœurs), les périphérique de stockage (disque dur, SSD, DVD), les périphérique d'entrées-sorties (clavier, souris, écran, imprimante), les périphériques de communication (réseau Ethernet et Wifi, Bluetooth, USB).
Chaque périphérique est piloté par un logiciel appelé « pilote » (driver).
- Les systèmes de fichiers (file system)
Un système de fichiers (FAT32, NTFS, Ext4, APFS) est une façon d'organiser et de formater les fichiers et dossiers sur un support de stockage (disque dur, clé USB,...)
- Les différents protocoles réseau (TCP-IP,...)
- Les sessions utilisateurs avec les droits associés

III) Les bibliothèques

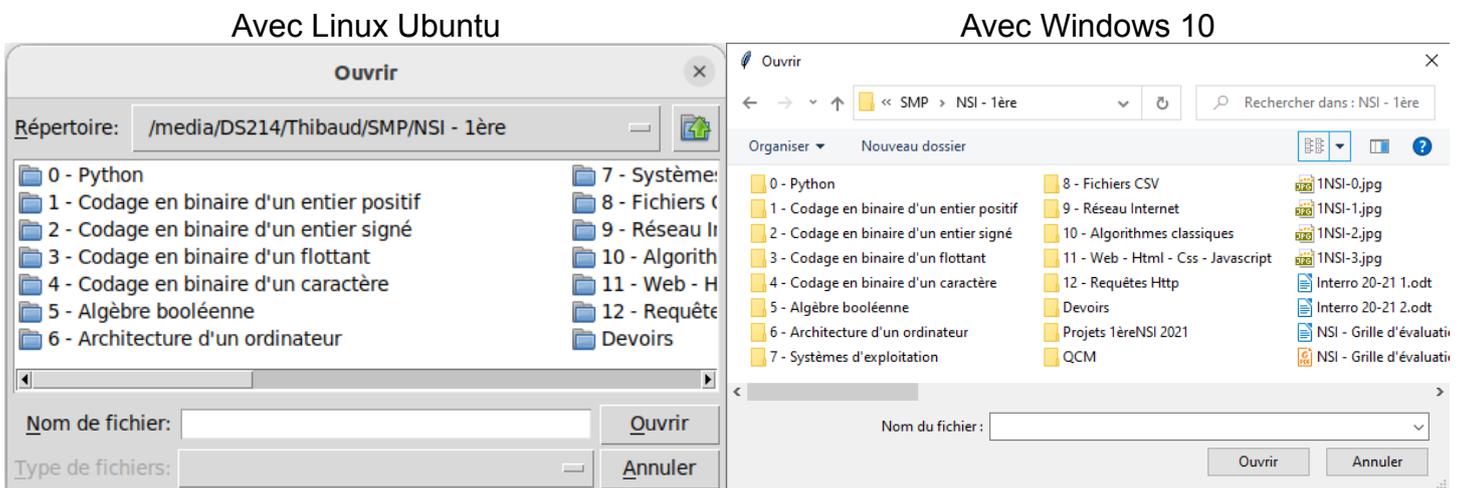
Les applications n'accèdent pas au noyau directement, mais via un ensemble de fonctions (appels système) regroupés dans des bibliothèques (API).

Exemple 1 : Si on exécute le code Python ci-dessous, l'interpréteur python va faire trois appels système : un pour ouvrir toto.txt, un pour lire son contenu et un pour le fermer. Le même code source python peut donc fonctionner sans modification sur des OS différents et avec des systèmes de fichiers différents. Le programmeur n'a pas besoin de connaître le fonctionnement interne du système de fichiers utilisé, ni même de savoir lequel est utilisé !

```
fichier = open("toto.txt")
texte = fichier.readlines()
fichier.close()
```

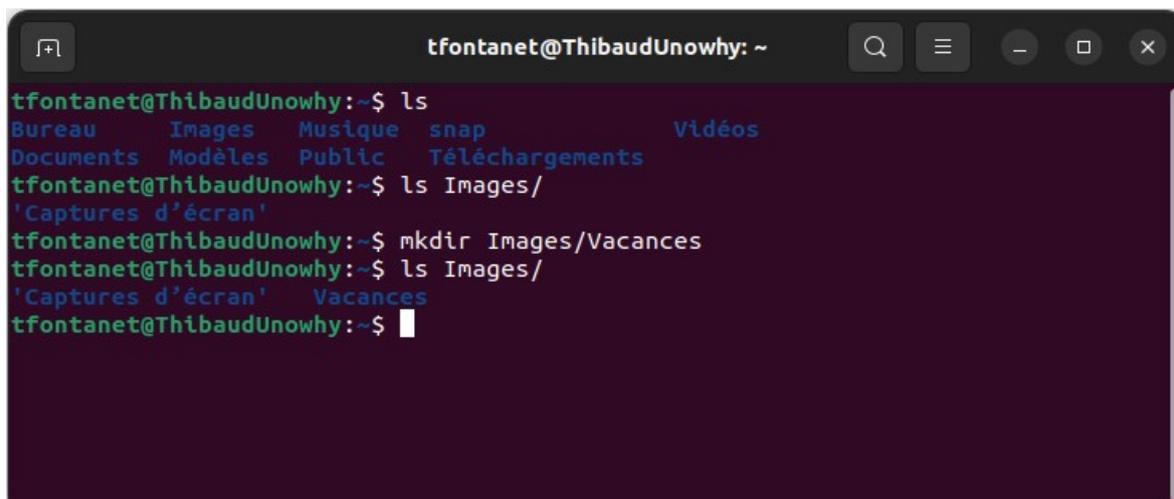
Exemple 2 : De même ce n'est pas l'application qui va créer elle-même les fenêtres, boutons et menus dont elle a besoin pour son interface utilisateur. Elle va plutôt faire des appels systèmes pour demander à l'OS de créer ces éléments pour elle.

Ainsi avec Python, la même fonction `tkinter.filedialog.askopenfilename()` n'a pas tout à fait la même allure selon l'OS utilisé.



IV) Interface en « Ligne de commande »

Les premières interfaces graphiques avec fenêtres, boutons, icônes,... se démocratisent dans les années 1980. Auparavant, l'utilisateur pouvait interagir avec les applications et l'OS uniquement via ce que l'on appelle « la ligne de commande ».



```
tfontanet@ThibaudUnowhy: ~  
tfontanet@ThibaudUnowhy:~$ ls  
Bureau  Images  Musique  snap      Vidéos  
Documents  Modèles  Public  Téléchargements  
tfontanet@ThibaudUnowhy:~$ ls Images/  
'Captures d'écran'  
tfontanet@ThibaudUnowhy:~$ mkdir Images/Vacances  
tfontanet@ThibaudUnowhy:~$ ls Images/  
'Captures d'écran'  Vacances  
tfontanet@ThibaudUnowhy:~$
```

Curieusement, 40 ans plus tard, la ligne de commande existe toujours ! En effet, de nombreuses tâches répétitives ou d'administration se font beaucoup plus efficacement avec la ligne de commande qu'avec une interface graphique. De nombreux serveurs ou matériels embarqués ne s'administrent qu'à distance en ligne de commande.

A noter que l'on peut rassembler plusieurs commandes dans un script et que ces scripts peuvent devenir de vrais programmes avec variables, boucles, tests, fonctions,... !