



FACULTÉ
DES SCIENCES
*Unité de formation
et de recherche*

UNIVERSITÉ
BRETAGNE
LOIRE

Système

Fabien GARREAU
sur la base du cours d'André Rossi

Université d'Angers
`fabien.garreau@univ-angers.fr`

L2 MPCIE, 2018–2019

Organisation

Page du cours de Système

<http://moodle.univ-angers.fr/course/view.php?id=10851>

Modalités de contrôle des connaissances

- Une épreuve écrite, pas de TP

Objectif du cours de système

- Connaître les grands principes de fonctionnement d'un système
- Comprendre pourquoi ils sont tels qu'ils sont aujourd'hui
- Savoir comment le système répond aux objectifs qui lui sont assignés

Bibliographie



Andrew Tanenbaum

Systèmes d'exploitation

3^{ème} édition

Pearson Education, 2008



Abraham Silberschatz et Peter Galvin

Principes des systèmes d'exploitation

4^{ème} édition

Addison-Wesley France, 1994



Samia Bouzefrane

Les systèmes d'exploitation

Sciences sup, Informatique

Dunod, 2003

Plus de références sous <http://moodle.univ-angers.fr/course/view.php?id=10851>

Les deux fonctions principales d'un système d'exploitation

En informatique, un système d'exploitation (OS - Operating System) est un **ensemble de programmes** qui dirige l'utilisation des ressources d'un ordinateur par des logiciels applicatifs.

Un système d'exploitation a deux fonctions principales

❶ **Proposer une abstraction du matériel au programmeur**

L'architecture est assez fastidieuse à programmer, en particulier les **E/S**. La solution est de proposer une **abstraction des ressources**.

Le système d'exploitation sert d'interface entre l'homme le matériel physique (**hardware**) avec lequel il communique directement.

❷ **Gérer les ressources matérielles de l'ordinateur**

Les deux fonctions principales d'un système d'exploitation

En informatique, un système d'exploitation (OS - Operating System) est un **ensemble de programmes** qui dirige l'utilisation des ressources d'un ordinateur par des logiciels applicatifs.

Un système d'exploitation a deux fonctions principales

- ➊ **Proposer une abstraction du matériel au programmeur**
- ➋ **Gérer les ressources matérielles de l'ordinateur**

L'OS est le chef d'orchestre : empêche que deux programmes écrivent en même temps au même endroit, gère les erreurs lors de l'exécution d'une application (ferme un programme si celui-ci est bloqué).

Il gère **le partage des ressources** dans le temps (le temps d'utilisation du CPU est partagé entre les utilisateurs) et dans l'espace (la mémoire est partagée entre différents processus)

Où trouve-t-on les systèmes d'exploitation ?

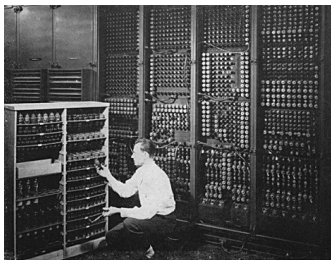
On trouve un système d'exploitation (*Operating System*, OS) dans de nombreux systèmes électroniques ou informatiques : les ordinateurs, où ils sont apparus, mais aussi les smartphones, box, bornes wifi, cartes électroniques sur systèmes embarqués, appareils électroménagers.

OS célèbres :

- Microsoft DOS
- Microsoft Windows
- UNIX
- GNU Linux
- Mac OS (Apple)
- OS/2 (IBM)
- Android (Google)
- iOS (Apple)
- Il existe des OS temps-réel
 - Ordinateur de bord, pilote automatique, automates programmables
- Un OS n'est pas toujours nécessaire (lave-linge, four micro-ondes)

Première génération (1945–1955) : tubes à vide

Les **tubes à vide** sont la seule technologie disponible (le transistor n'est inventé qu'en 1947). L'interface est constituée de tableaux d'interrupteurs. Programmation en langage machine absolu.

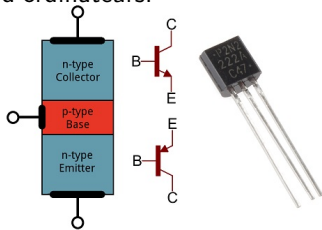


Aucun langage de programmation n'existait (pas même l'assembleur),
pas de système d'exploitation.

Vers 1950, utilisation de **cartes perforées.**

Seconde génération (1955–1965) : le temps des transistors

Grâce aux transistors, la **fiabilité augmente**, le retour sur investissement devient possible, les (très grandes) entreprises s'équipent d'ordinateurs.

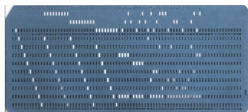


Une définition plus nette des rôles de concepteurs, constructeurs, programmeurs et personnels de maintenance.

Un **mainframe** occupe une pièce climatisée, et coûte plusieurs millions de dollars. Les mainframe sont les ancêtres des **fermes de calcul** (cluster) d'aujourd'hui.

Seconde génération (1955–1965) : traitement par lots

Le programmeur écrit son programme en **assembleur ou Fortran** sur une ou plusieurs **cartes perforées**. Un opérateur rassemble toutes les cartes des utilisateurs pour constituer **un lot**, le soumet à un premier "petit" ordinateur muni d'un lecteur de cartes qui écrit tous les programmes à exécuter sur une bande magnétique d'entrée.



La bande magnétique d'entrée est montée manuellement sur l'ordinateur central (**IBM 1401**). L'ordinateur exécute tous les programmes, il écrit le résultat sur la bande magnétique de sortie au fur et à mesure.

Une fois le traitement du lot terminé, un opérateur humain prend la bande magnétique de sortie et la monte sur un "petit" ordinateur qui imprime les résultats sur papier.

Les **mainframes** se distinguent par leur puissance de calcul et leur aptitude à gérer **un très grand nombre d'E/S**.

Troisième génération (1965–1980) : circuits intégrés

La disponibilité des circuits intégrés provoque une forte diminution du coût des ordinateurs, et permet l'augmentation des performances.

Fin du découpage entre petits ordinateurs spécialisés dans les interfaces et gros ordinateurs orientés calculs, mais une machine unique. Le premier est l'IBM 360.

IBM a souhaité que le code écrit pour une machine soit utilisable sur les suivantes, y compris l'OS appelé OS/360. Grande diversité d'usage (banque assurance, calculs) implique qu'OS/360 est très complexe, beaucoup de correctifs, mais toujours beaucoup de bugs. Malgré tout une innovation majeure : la multiprogrammation (= le multitâche). L'accès à une donnée sur bande magnétique est très long, et l'UC (Unité Centrale de traitement) des ordinateurs de 2ème génération attendait la donnée pour continuer. Gaspillage de temps, et sous-utilisation d'un matériel cher.

Troisième génération (1965–1980) : multiprogrammation

La multiprogrammation avec OS/360 consiste à **charger plusieurs jobs en mémoire centrale**, et à exécuter l'un d'entre eux. Quand l'UC ne peut poursuivre le job en cours car une donnée est manquante, elle est attribuée à un autre job. OS/360 protège la mémoire contre **l'intrusion de jobs concurrents**.

Autre innovation d'OS/360 : le **spoole** : on stocke sur disque dur les jobs écrits sur carte au fil de leur arrivée. Dès qu'un job est terminé, on le remplace en mémoire par un job sur le disque. On utilise également le spoole en sortie pour l'impression des résultats.

Pour rentabiliser encore l'utilisation de l'ordinateur, le **temps partagé** est introduit par le Massachusetts Institute of Technology : chaque utilisateur dispose d'un terminal pour soumettre des jobs. Il n'est plus nécessaire d'attendre plusieurs heures la fin du traitement d'un lot, on parle de **mode interactif**.

Parallèlement, les mini-ordinateurs apparaissent vers 1961, mais restent chers (**120 000 dollars pièce**). UNIX a été créé pour ces machines.

Quatrième génération (de 1980 à nos jours) : l'essor

Les circuits électroniques se miniaturisent (circuits LSI : Large Scale Integration, puis VLSI, où V signifie Very), les micro-ordinateurs deviennent abordables pour les particuliers. En 1980, IBM commercialise le PC (Personal Computer) et cherche un OS, Bill Gates propose alors MS-DOS.

Depuis, la miniaturisation, la baisse des coûts et l'augmentation des performances ont conduit à l'essor de l'informatique ubiquitaire (pervasive computing).