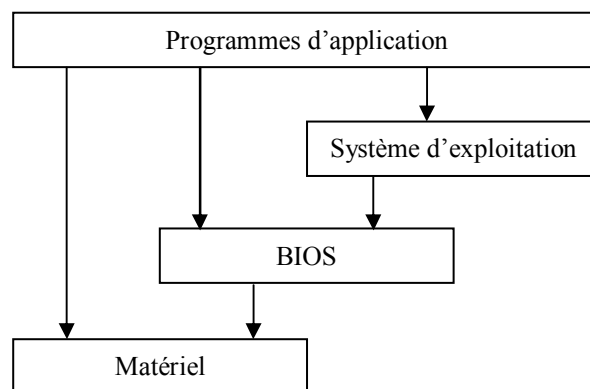


# Systeme d'exploitation : objectifs et fonctions

## 1. Introduction

Au départ la machine ne comporte que le matériel (unité centrale, mémoire, périphérique d'E/S, ...). En l'absence de l'aide d'un logiciel quelconque, les opérations de chargement et d'exploitation d'un programme et la gestion du matériel sont sous la dépendance totale du contrôle humain.

La manière usuelle d'accéder au matériel consiste à utiliser un intermédiaire qui propose des services spécialisés. Cet intermédiaire peut être le BIOS ou un système d'exploitation qui représente des interfaces logicielles pour gérer le matériel.



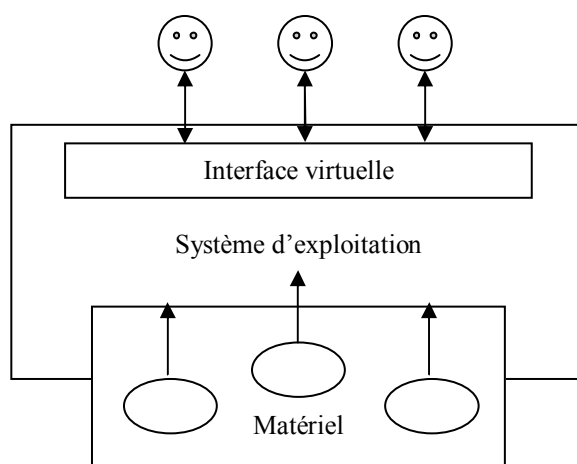
On s'intéresse ici au système d'exploitation qui désigne l'ensemble des programmes qui réalisent l'interface entre le matériel de l'ordinateur et les utilisateurs.

## 2. Objectifs

On souligne deux objectifs fondamentaux du système d'exploitation.

D'une part, il construit sur machine physique tel qu'elle est livrée par le constructeur, une machine virtuelle plus facile pour l'emploi et plus conviviale (ergonomique) pour l'utilisateur.

D'autre part, il prend en charge la gestion complexe des ressources de l'ordinateur (processeur, mémoire, périphériques,...), en optimisant l'utilisation et permettant le partage entre les utilisateurs.



### 3. Historique

L'histoire de l'informatique est très brève (les ordinateurs sont nés avec la deuxième guerre mondiale) et pourtant elle a connue de grandes évolutions à leurs apparitions. Les ordinateurs étaient très coûteux et réservés aux grandes entreprises. Ces ordinateurs centraux sont rapidement devenus un auxiliaire d'administration et ils se sont diffusés dans les différents services (financier, administratifs,...).

Pour rendre l'informatique plus adaptée et plus abordable, des fabricants se sont mis à fournir des minis ordinateurs. Les ordinateurs fonctionnent avec des systèmes d'exploitation qui leur étaient propres, à chaque machine ou à chaque constructeur (MVS pour IBM, VMS pour DEC,...).

Aujourd'hui l'informatique, aussi bien dans l'entreprise que dans la recherche ou l'enseignement, utilise des machines plus petites, fonctionnant avec des systèmes d'exploitation à caractère universel. Parmi ces systèmes d'exploitation deux se distinguent particulièrement, un système mono-utilisateur (Windows) et un autre multi-utilisateur et multitâches (Unix).

Unix est le plus ancien et celui qui offre le plus de richesse, le plus d'homogénéité et le plus de souplesse. Il dispose dans les versions standard, d'extension pour des réseaux et pour le graphique.

L'étude des systèmes d'exploitation forme une part très importante de l'informatique comme discipline et à la différence de ces autres domaines c'est une part qui lui est propre.

### 4. Fonctions

On peut diviser les systèmes d'exploitation classiques en 4 parties :

#### 4.1. Les processus

Les processus qui correspondent à l'exécution des programmes. Ces processus pouvant s'exécuter simultanément dans un système multitâches.

Le système a pour fonction de les créer, les gérer, les synchroniser ainsi que de leur permettre de communiquer entre eux.

#### **4.2. La gestion de la mémoire**

Elle permet de transférer les programmes et les données nécessaires à la création des processus, d'un support secondaire (ex : un disque) vers un support centrale où aura lieu l'exécution du processus.

Le système devra garder la trace des parties utilisées et libres de la mémoire, ainsi que gérer les transferts entre les mémoires principales et secondaires.

#### **4.3. Le système de fichiers**

Le système de fichiers qui offre à l'utilisateur une vision homogène et structurée des données et des ressources. Le système gère la création des fichiers, leur destruction, leur correspondance avec les dispositifs physiques ainsi qu'un certain nombre d'autres caractéristiques tel que la sécurité.

#### **4.4. Les entrées sorties**

Les entrées sorties correspondent au mécanisme qu'utilisent les processus qui communiquent avec l'extérieur. Ces entrées sorties font l'appel aux couches les plus proches du matériel et dont le système tente de masquer les particularité aux utilisateurs.

Les systèmes d'exploitations modernes intègrent d'autres caractéristiques. Ces dernières concernent deux évolutions majeures des systèmes informatiques. La première est l'interconnexion des différentes machines et des différents systèmes par des réseaux locaux ou étendus. La seconde est la disparition des écrans textes et leur remplacement par des dispositifs à fenêtres multiples disposant des propriétés graphiques.

Le système d'exploitation correspond à l'interface entre les applications et le matériel. Le programmeur d'application n'aborde que rarement sinon jamais son code interne. Il l'utilise par l'intermédiaire d'appel système.

Les appels systèmes sont souvent accessibles à partir d'un langage de programmation en C dans le système Unix par exemple. Ces appels système permettent d'effectuer la plus part des opérations sur les entités du système d'exploitation et par exemple de créer et détruire des processus, des fichiers, de réaliser des entrées sorties,...

## 5. Problématique

Pour que les programmes puissent s'exécuter de façon portable et efficace, il faut pouvoir gérer simultanément :

- La multiplicité des différentes ressources
- La complexité des composants de chacune d'elles, qui demandent la prise en compte de nombreux détails

On peut appréhender ces problèmes à travers un exemple

### Exemple

Une machine multi-utilisateur fournit un service d'impression qui peut être utilisé par n'importe quel programme s'exécutant sur la même machine. Pour cela il faut :

- Pouvoir verrouiller l'accès à l'imprimante, afin que les flots de caractères produits par les programmes désirant imprimer ne s'entrelacent pas sur les papiers
- Gérer des tampons d'impression, afin que les programmes puissent reprendre leur travail sans devoir attendre la fin effective de l'impression

Le problème ici est de gérer l'accès à une ressource coûteuse. A tout instant il faut :

- Gérer l'accès concurrent à cette ressource
- Pouvoir accorder l'usage exclusif à cette ressource
- Éviter les conflits entre les programmes ou entre les usagées