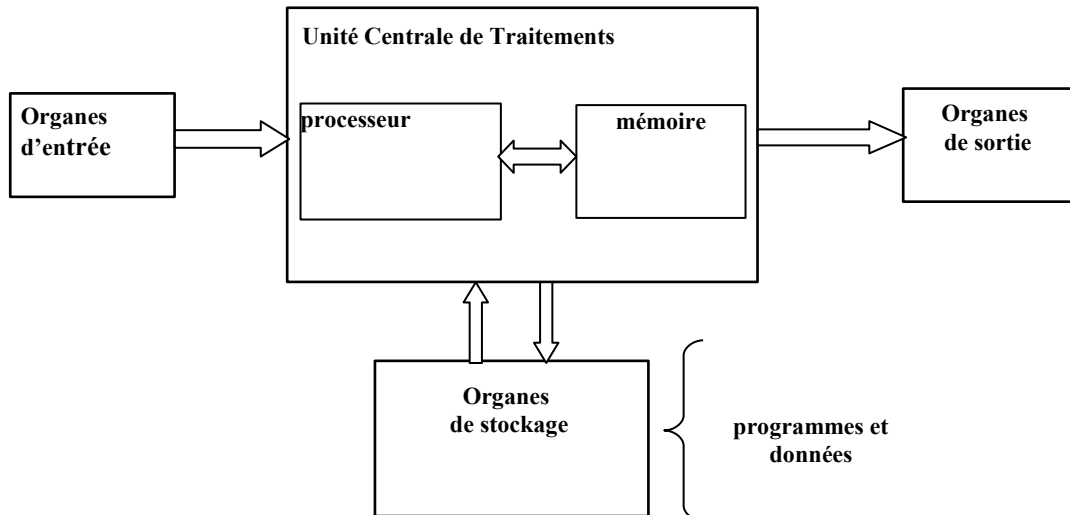


**Systèmes d'exploitation : Notion essentielles**

**I. : Qu'est-ce qu'un Système d'Exploitation ?**

Un ordinateur est principalement composé de circuits électroniques, de dispositifs divers d'entrée / sortie de données et de dispositifs de stockage de données.

**Schéma fonctionnel d'un ordinateur**

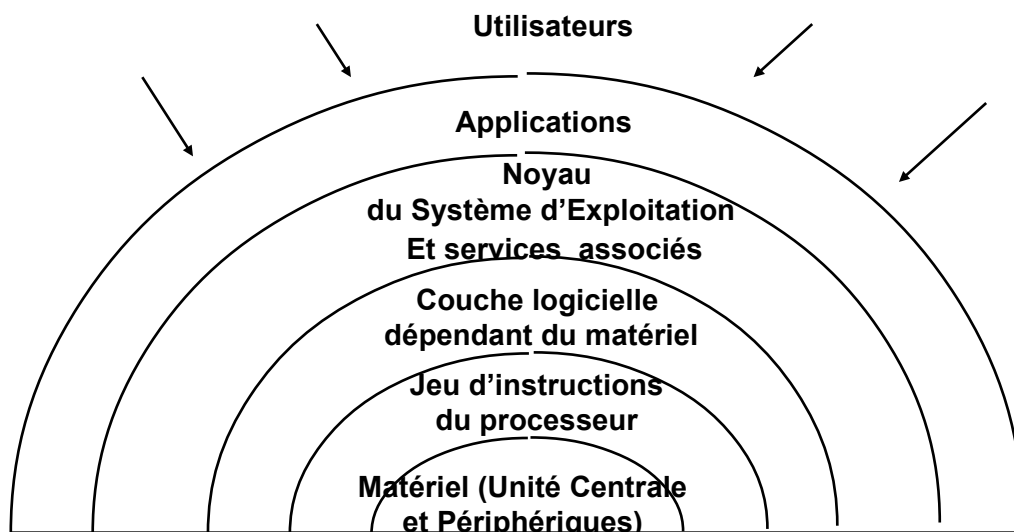


Son intérêt est d'être le support à l'exécution de programmes ou logiciels. Un logiciel qui peut intéresser un utilisateur, c'est par exemple un traitement de texte.

**Mais pour que ce logiciel puisse "tourner" sur la machine, il doit utiliser les services d'autres programmes qui composent le système d'exploitation :**

- chargement du programme depuis l'unité de stockage dans la mémoire centrale
- lancement ou démarrage du programme
- impression de documents
- stockage de documents sur unité de disque

Il est commode de représenter l'interaction entre les différents éléments (matériels et logiciels) par un **modèle en couches** (les modèles en couche sont très utilisés en informatique).



**Un ordinateur = une machine universelle** conçue pour servir à de multiples usages.

Des couches logicielles successives permettent une spécialisation de la machine, l'adaptant ainsi à ses besoins.

Nous parlerons alors de **machine virtuelle**, exprimant par là que nous ne percevons la machine qu'à travers les fonctionnalités qui nous sont proposées.

Pour les applications, le système d'exploitation permet un **accès transparent au matériel** : dispositifs d'entrées-sorties de données, unités de stockage, imprimantes et tout périphérique connecté à l'ordinateur.

**Exemples:**

une application n'a pas à savoir si elle écrit sur une partition de disque formatée en NTFS ou une clé USB formatée en fat32

**Définition**

Le système d'exploitation est l'ensemble des programmes, généralement fournis avec la machine, permettant de faciliter l'emploi des ressources matérielles et logicielles du système informatique par les applications et par les utilisateurs.

**II. Fonctions principales du système d'exploitation. :**

- Permettre le lancement et l'exécution d'un programme
- Gérer l'ensemble des ressources du système : Processeur(s), Mémoire Centrale, Unités de Stockage, Périphériques d'entrée-sortie.
- S'occuper des problèmes d'entrée-sortie de données (clavier, écran, disques, disquettes, imprimante, souris...). Pour cela le système d'exploitation fait appel à des **pilotes** ou **drivers** qui sont des petits programmes conçus pour interagir avec un matériel particulier (imprimante, carte graphique, ...).
- **Gérer la communication** réseau (prise en charge des couches réseaux), ce qui est essentiel aujourd'hui et le partage de données sur les réseaux.
- Assurer la stabilité, la sécurité, la fiabilité du système

Le Système d'Exploitation est de façon globale **composé d'un noyau**, prenant en charge les fonctions essentielles, **et d'utilitaires ou services ou modules**, qui sont en fait des programmes comme les programmes utilisateurs.

Parmi les services offerts par un Système d'exploitation, il en est deux qui présentent une importance particulière pour l'utilisateur :

**L'interpréteur de commandes :**

qui permet à l'utilisateur d'interagir avec le système d'exploitation (de lui donner des instructions). Cet interpréteur de commandes peut être en mode caractère (la ligne de commandes sous Windows, le shell sous Linux) ou en mode graphique (**GUI** : **Graphic User Interface**)

Aujourd'hui, l'interface graphique a prise une place prépondérante dans l'évaluation des systèmes d'exploitation pour ordinateurs personnels. C'est ce que l'utilisateur voit du système d'exploitation.

**le Système de Gestion de Fichiers (le SGF) :**

qui a la charge de gérer le stockage des données

### III. Lancement de programmes

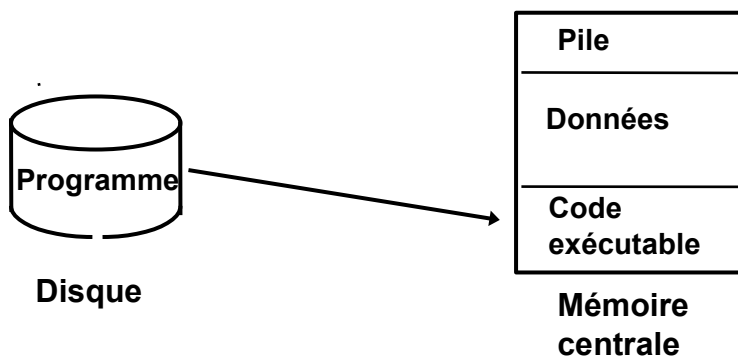
Une des fonctions du SE, parmi les plus utilisées directement par l'utilisateur : **le lancement de programmes**.

- **chargement du programme** depuis l'unité de stockage dans la mémoire centrale (sauf pour les programmes stockés en mémoire morte). Fonctions nécessaires pour ce chargement :
  - gestion des entrées-sorties disque
  - gestion du stockage sur disque (**S**ystème de **G**estion de **F**ichiers)
  - gestion de la mémoire
- **lancement du programme** : le SE donne instruction au processeur d'exécuter le programme, mais néanmoins le SE doit garder le contrôle de la machine.

### IV. Notion de processus

#### 1.) Définition

Lancement et exécution d'un programme :



#### processus en mémoire centrale

Un programme qui "tourne" dans l'ordinateur, c'est-à-dire qui est exécuté par l'Unité Centrale, est appelé **un processus**.

Chaque processus en mémoire centrale dispose de son propre environnement, créé lors du lancement du programme, c'est-à-dire d'une zone de mémoire où l'on trouve **le code exécutable**, **les données manipulées par le programme** et **une pile de données**.

#### ***Espace d'adressage des processus***

*Chaque processus possède un espace d'adressage, c'est-à-dire un ensemble d'adresses mémoires dans lesquelles il peut lire et écrire.*

*Cet espace est divisé en trois parties :*

*le segment de texte (le code du programme) ;*

*le segment de données (les variables) ;*

*la pile.*

Le processus est ce que connaît le Système d'Exploitation **d'un programme**. C'est l'unité de logiciel à laquelle le SE doit attribuer des ressources : de la place en mémoire centrale, du temps CPU pour exécution, les périphériques sollicités.



Ce qui caractérise un processus :

- **l'environnement en mémoire centrale** du processus
- **l'état du processeur** à un moment donné, c'est-à-dire le **contenu des registres du processeur**.

### **Très important :**

**Chaque processus exécuté par le système d'exploitation se voit attribuer une zone mémoire protégée, dans laquelle aucun autre processus ne peut accéder.**

## **2 ) Les threads :**

Plus précisément, un programme en exécution peut consister en plusieurs processus. On parle également de **threads** et de **programmes multi-threads**. Ce sont en quelque sorte des mini processus, créés par le processus principal, dépendant de lui, partageant des données avec lui mais dont l'exécution se poursuit en parallèle : exemple sous Word le correcteur orthographique fonctionne dans un thread indépendant.

Grâce aux threads, dans une même application plusieurs tâches peuvent se dérouler en parallèle.

## **3 ) Systèmes d'Exploitation mono-tâche / multi-tâches**

**Dans un SE mono-tâche**, un seul processus est en exécution à un moment donné dans l'Unité Centrale (exemple Ms-Dos).

Le processus dispose de toutes les ressources du système.

**Dans un SE multi-tâches**, plusieurs processus sont en exécution à un moment donné dans l'Unité Centrale.

### **Parallélisme et pseudo-parallélisme**

*La base des ordinateurs modernes, c'est le parallélisme. Il est désormais inconcevable qu'un ordinateur ne puisse exécuter plusieurs programmes en même temps.*

*Du point de vue matériel, le processeur passe d'un programme à un autre en quelques millisecondes, ce qui va donner à l'utilisateur une impression de simultanéité. C'est le pseudo-parallélisme, à différencier avec le véritable parallélisme des systèmes multiprocesseurs.*

*Les processus sont la base du pseudo-parallélisme.*

Chaque processus dispose de son propre environnement en mémoire centrale.

### **Mais attention :**

S'il n'y a qu'un processeur dans l'Unité Centrale , **à un instant donné, il ne peut y avoir qu'un processus en exécution par l'UC**. Il y a donc contradiction avec le fait de dire qu'il y a plusieurs processus en exécution dans l'UC. **Qu'entend-on alors par multi-tâches ?**

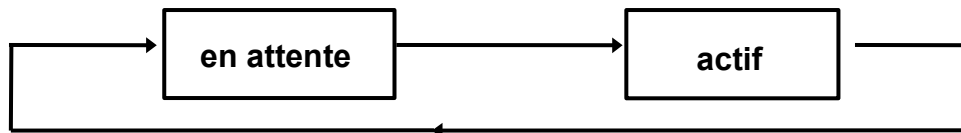
Un processus peut être en fait se trouver dans deux états différents :

**actif** ou **en attente**.

Pendant qu'un processus est actif dans l'UC (traitement des instructions par le processeur), les autres processus sont en attente.

C'est au Système d'exploitation d'allouer la ressource processeur à un processus en attente et donc de mettre en attente celui qui était actif. C'est une des fonctions, très importante, du Système d'exploitation).

### un processus



#### **Ordonnancement des processus**

Lorsqu'un processus est lancé, le système doit gérer la mémoire et l'allocation du processeur lui étant accordée. Il fait appel à l'ordonnanceur (scheduler en anglais).

Un système d'exploitation est préemptif lorsqu'il peut arrêter à tout moment n'importe quelle application pour passer à la suivante (exemple : Windows 7 et GNU/Linux sont des systèmes préemptifs).

multitâche préemptif : le système d'exploitation garde le contrôle et se réserve le droit de fermer l'application.

*Plusieurs processus peuvent se trouver simultanément en cours d'exécution (multiprogrammation et temps partagé), si un système informatique ne comporte qu'un seul processeur, alors, à un instant donné, un seul processus aura accès à ce processeur. En conséquence, un programme en exécution peut avoir plusieurs états.*

#### **États d'un processus :**

*Dans un système multitâches, plusieurs processus peuvent se trouver simultanément en cours d'exécution : ils se partagent l'accès au processeur.*

*Un processus peut prendre 3 états :*

- *Etat actif ou élu (running): le processus utilise le processeur.*
- *Etat prêt ou éligible (ready: le processus pourrait utiliser le processeur s'il était libre (et si c'était son tour).*
- *Etat en attente ou bloqué : le processus attend une ressource (ex : fin d'une entrée-sortie).*

*Le S.E choisit un processus qui deviendra actif parmi ceux qui sont prêts. Tout processus qui se bloque en attente d'un événement (par exemple l'attente de la frappe d'un caractère au clavier lors d'un scan) passe dans l'état bloqué tant que l'événement attendu n'est pas arrivé. Lors de l'occurrence de cet événement, le processus passe dans l'état prêt. Il sera alors susceptible de se voir attribuer le processeur pour continuer ses activités. Les processus en attente sont marqués comme bloqués dans la table des processus. Ils se trouvent généralement dans la file d'attente liée à une ressource (imprimante, disque, ...).*

*Le changement d'état d'un processus peut être provoqué par :*

- *un autre processus (qui lui a envoyé un signal, par exemple)*
- *le processus lui-même (appel à une fonction d'entrée-sortie bloquante, ...)*
- *une interruption (fin de quantum , terminaison d'entrée-sortie, ...)*

*La sortie de l'état actif pour passer à l'état prêt se produit dans le cas des ordonnancements préemptifs lorsqu'un processus plus prioritaire que le processus actif courant devient prêt.*

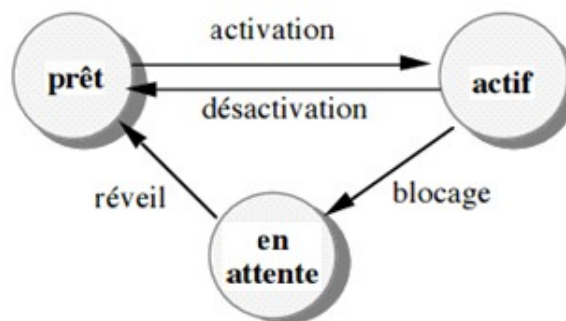
D'une manière générale :

- le passage de l'état actif à l'état prêt est provoqué par le système en fonction de sa politique d'ordonnancement (fin de quantum, préemption du processus actif si un processus plus prioritaire devient prêt dans le cas des politiques préemptives, ...),
- le passage de l'état actif à l'état bloqué est provoqué par le programme exécuté par le processus.

### Transitions entre états

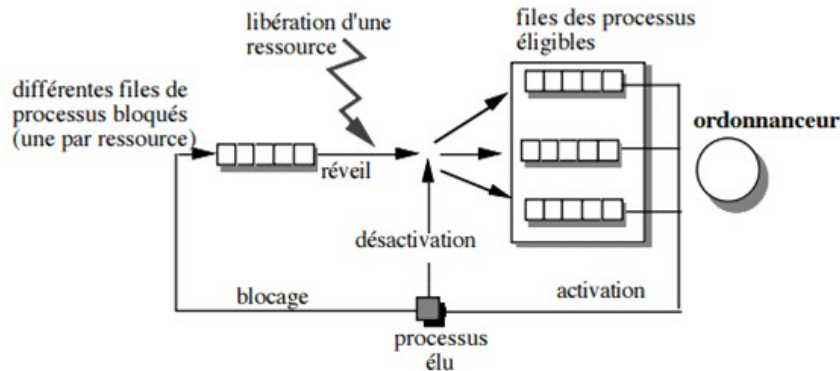
- Les transitions entre états provoquent le passage d'un état à un autre :

Activation	prêt -> actif.
Désactivation (ou préemption, ou réquisition)	actif -> prêt.
Mise en attente (ou blocage)	actif -> en attente.
Réveil	attente -> prêt.



# Ordonnancement

- Dans un système multitâches, le système d'exploitation doit gérer l'allocation du processeur aux processus. On parle d'*ordonnancement* des processus.

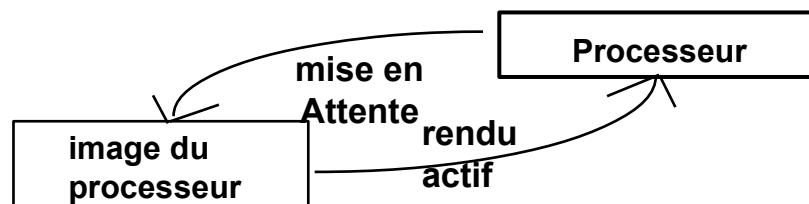


- Il existe différentes politiques d'allocation :
  - avec ou sans priorité
    - o sans : premier arrivé premier servi (*first come, first served* : FCFS)
    - o avec :
      - la priorité peut être fixe ou dynamique
      - il peut y avoir préemption, ou non

## Point important :

Un processus en attente doit conserver l'image de l'état du processeur (état des registres) au moment où il est mis en attente. Et lorsque le processus redevient actif, cette image de l'état du processeur doit être rechargé dans le processeur.

Cette image du processus est stockée dans la partie "pile de données" de l'environnement du processus en mémoire centrale (c'est l'un des rôles de cette pile de données).





*ETUDE PROCESSUS STATIONS : voir doc gestionnaire de taches*

### **III. Gestion de la mémoire vive**

Ce rôle du Système d'exploitation est essentiel :

- un programme, pour s'exécuter, doit être chargé en mémoire vive
- un processus (le programme en exécution) est caractérisé par un environnement en mémoire vive qui est propre à ce processus.

On peut distinguer les tâches suivantes dans la gestion de la mémoire vive par le Système d'exploitation :

- trouver un espace libre pour charger un programme
- traiter les échanges mémoire vive – unités de stockage (disques durs ..)
- créer un environnement en mémoire pour un processus
- maintenir à jour une table des processus avec leur occupation mémoire
- assurer la protection de l'environnement mémoire d'un processus par rapport aux autres processus : un processus ne doit pas accéder (ni en lecture, ni en écriture) à l'espace mémoire d'un autre processus
- récupérer la mémoire occupée par un processus lorsqu'il est terminé

#### **Mémoire virtuelle**

Généralement la mémoire vive disponible ne suffit pas pour accueillir l'environnement d'exécution de tous les processus. Sans dispositif particulier, à la demande de lancement d'un programme, s'il n'y avait plus assez de place en mémoire vive, le Système d'exploitation refuserait le chargement et renverrait un message d'erreur.

Pour pallier à ce problème les Systèmes d'exploitation mettent en œuvre un dispositif de **mémoire virtuelle**.

Dans ce dispositif, une partie du disque dur est utilisée comme annexe de la mémoire vive. Lorsqu'il n'y a plus assez de place en mémoire vive pour l'accorder à un processus qui en a besoin, le Système d'exploitation stocke une partie du contenu de la mémoire vive dans cet espace réservé du disque dur et libère la zone mémoire correspondante.

Bien sûr quand le processus, qui utilisait cette zone mémoire qui a été déchargée sur le disque dur, redevient actif, le Système d'exploitation doit ramener en mémoire vive cette zone et pour cela décharge sur disque dur la zone mémoire d'un autre processus.

On assiste donc à un va-et-vient entre la mémoire vive et le disque dur :

c'est ce qu'on appelle **le swap**. On dit que ça swape.

Lorsque la mémoire vive, présente sur la machine, est insuffisante par rapport aux applications qui tournent simultanément, le Système d'exploitation est obligé d'opérer des va-et-vient incessant entre la mémoire vive et le disque dur. Cela dégrade énormément les performances.

**Fichier d'échange** sous Windows : c'est la zone de disque qui est utilisée par Windows pour ce processus de mémoire virtuelle. C'est un fichier qui s'appelle **pagefile.sys**

**Partition de swap** sous Linux: pour ce système d'exploitation cette zone de mémoire virtuelle est prévue sous la forme d'une **partition** , qui doit être définie à l'installation de Linux.